



CRONICAS: RECONOCIMIENTO DE PATRONES

Jose Aguilar

Reconocimiento de Eventos

- **Entrada:**

- La representación simbólica del tiempo estampillado: eventos de bajo nivel (low-level events, LLE).
- LLE viene desde diferentes fuentes/sensores.
- Muchas LLE.

- **Salida:**

- Alto nivel de eventos (HLE), combinaciones de LLE y/o HLE.
- Humanos entiende más fácil HLE que LLE.

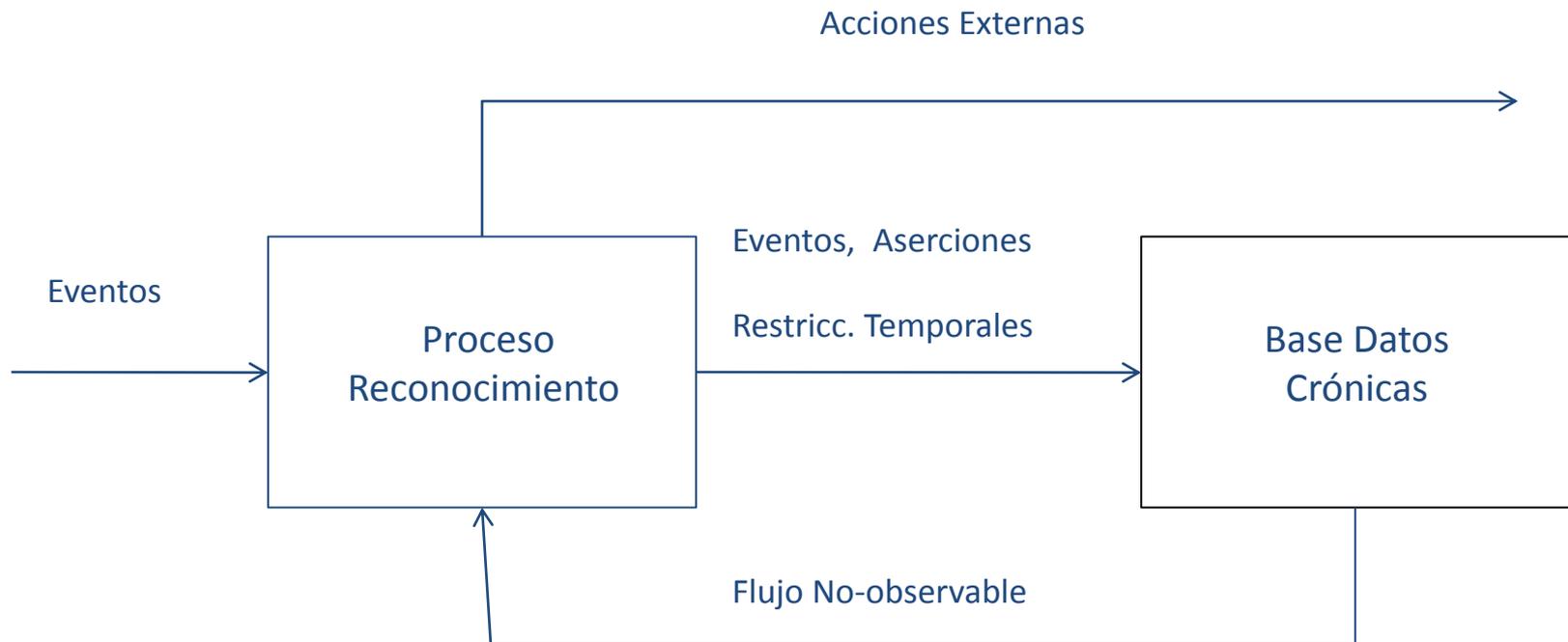
- **Reconocimiento de Eventos puede ser:**

- On-line (run-time).
- Off-line (retrospective)

Reconocimiento en las Crónicas

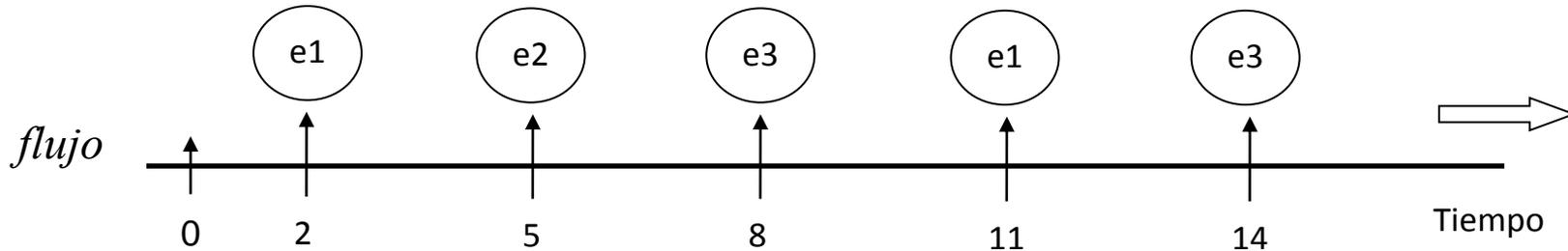
- CRP es un método de razonamiento temporal que realiza un **reconocimiento temporal en línea de los patrones de eventos en un flujo de eventos** viniendo o sensados desde dispositivo que observan algún sistema o el medio ambiente de interés.
- **Estos patrones, llamados crónicas, representan una posible evolución** (normal o anormal) del estado del sistema observada.

Reconocimiento en las Crónicas



Reconocimiento en las Crónicas

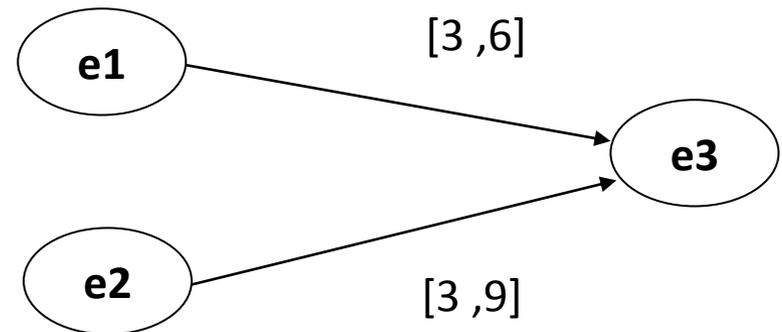
Reconocer es explicar los eventos de entrada (flujo) con la ayuda de patrones temporales (S), respetando las limitaciones del dominio (T)



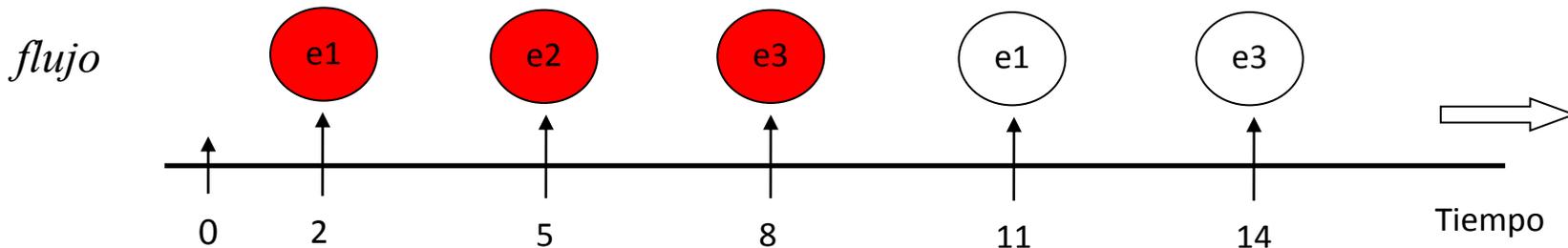
$$C = \{S, T\}$$

$$S = \{e1, e2, e3\}$$

$$T = \{t1 < t3 \text{ avec } 3 \leq t3 - t1 \leq 6 \}$$
$$\{t2 < t3 \text{ avec } 3 \leq t3 - t2 \leq 9 \}$$



Reconocimiento en las Crónicas

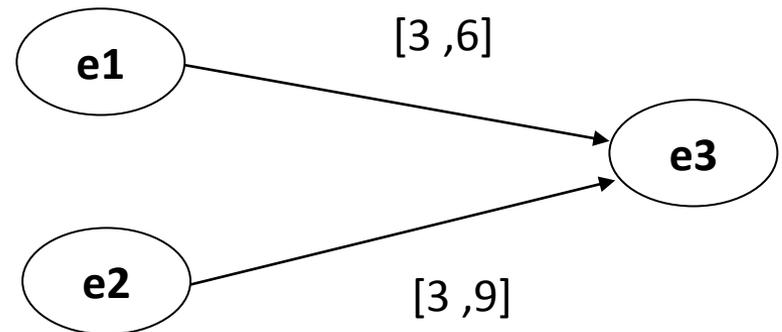


Primer Reconocimiento : Secuencia S-1

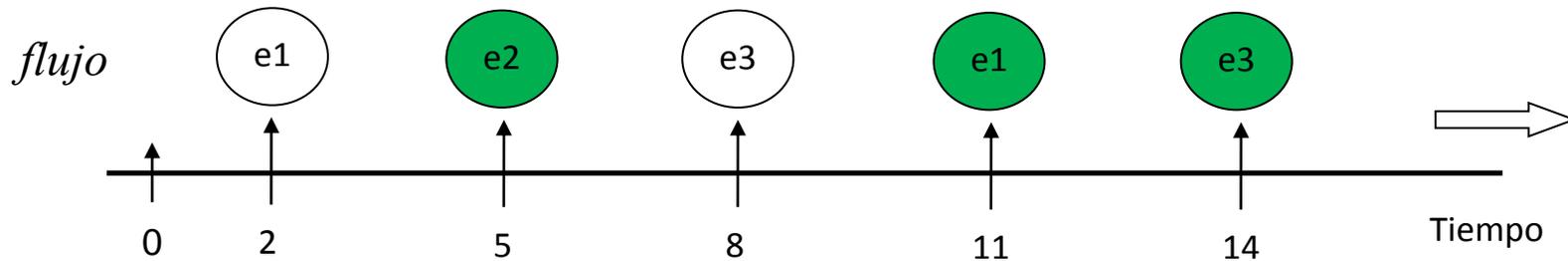
$$C = \{S, T\}$$

$$S = \{e1, e2, e3\}$$

$$T = \{t1 < t3 \text{ avec } 3 \leq t3 - t1 \leq 6 \}$$
$$\{t2 < t3 \text{ avec } 3 \leq t3 - t2 \leq 9 \}$$



Reconocimiento en las Crónicas

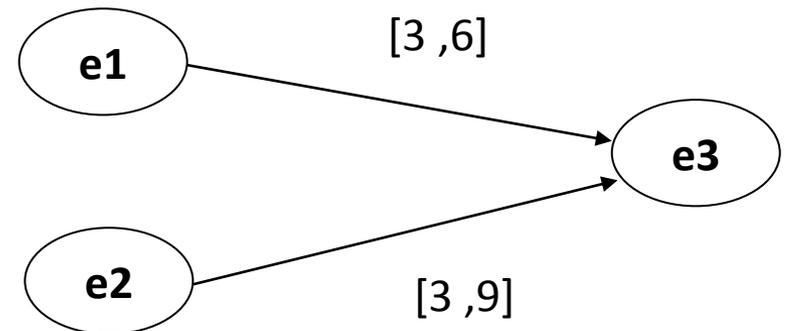


2do. Reconocimiento : Secuencia S-2

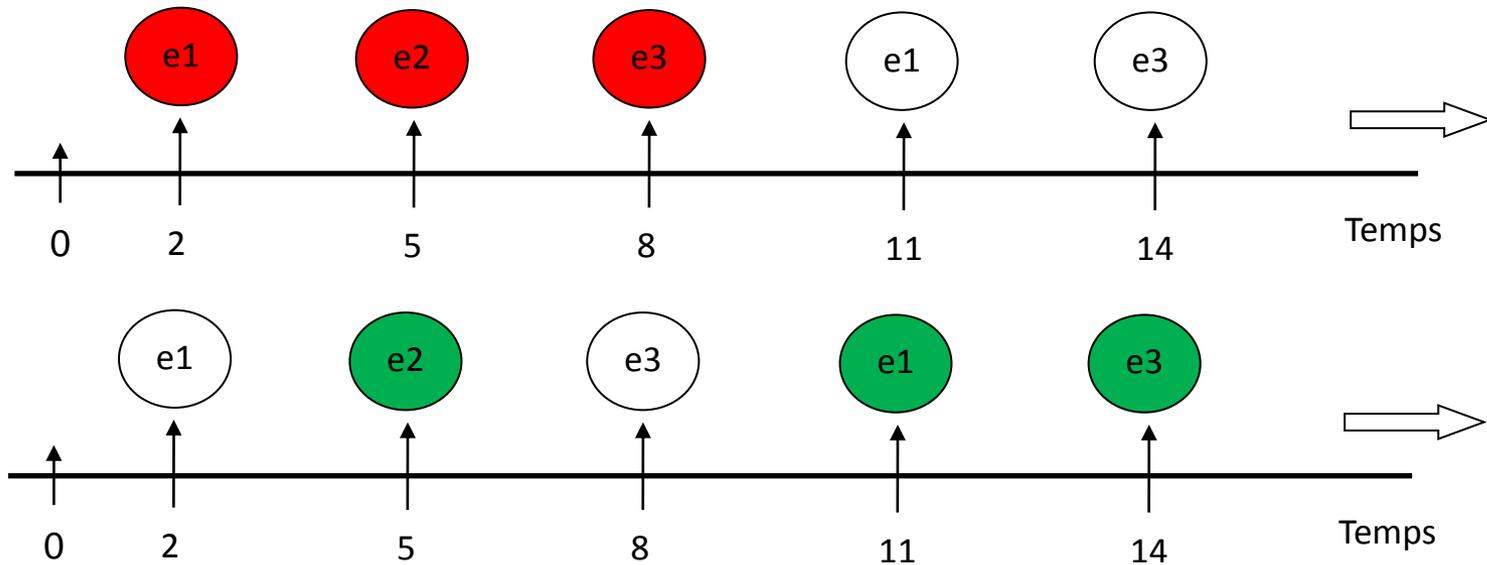
$$C = \{S, T\}$$

$$S = \{e1, e2, e3\}$$

$$T = \{t1 < t3 \text{ avec } 3 \leq t3 - t1 \leq 6\}$$
$$\{t2 < t3 \text{ avec } 3 \leq t3 - t2 \leq 9\}$$

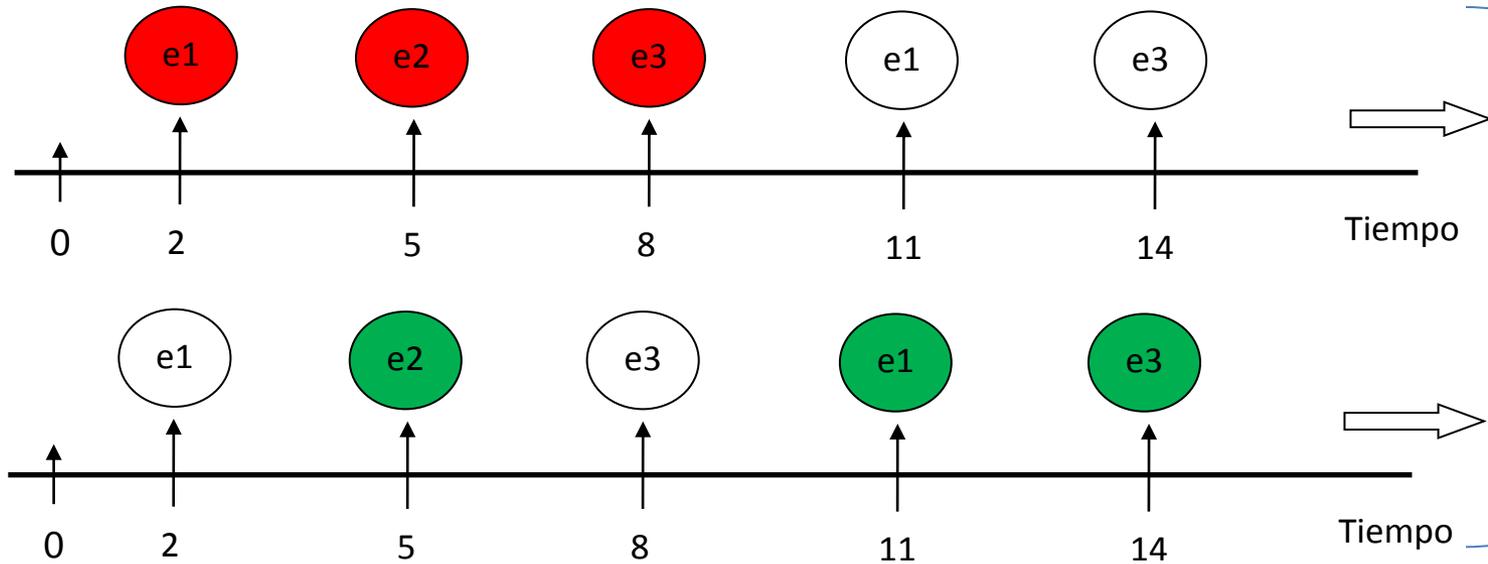


Reconocimiento en las Crónicas

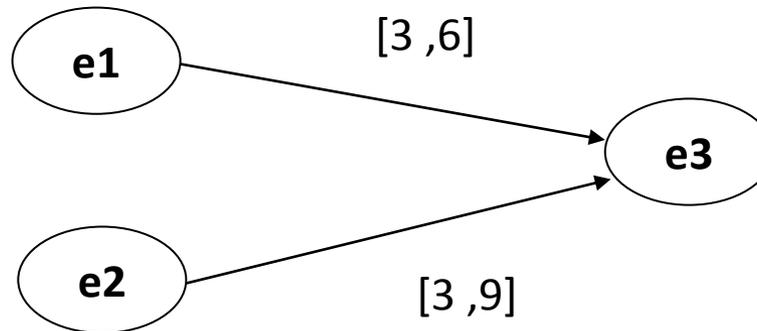


Las secuencias S-1 y S-2 son 2 instancias de la crónica C

Reconocimiento en las Crónicas

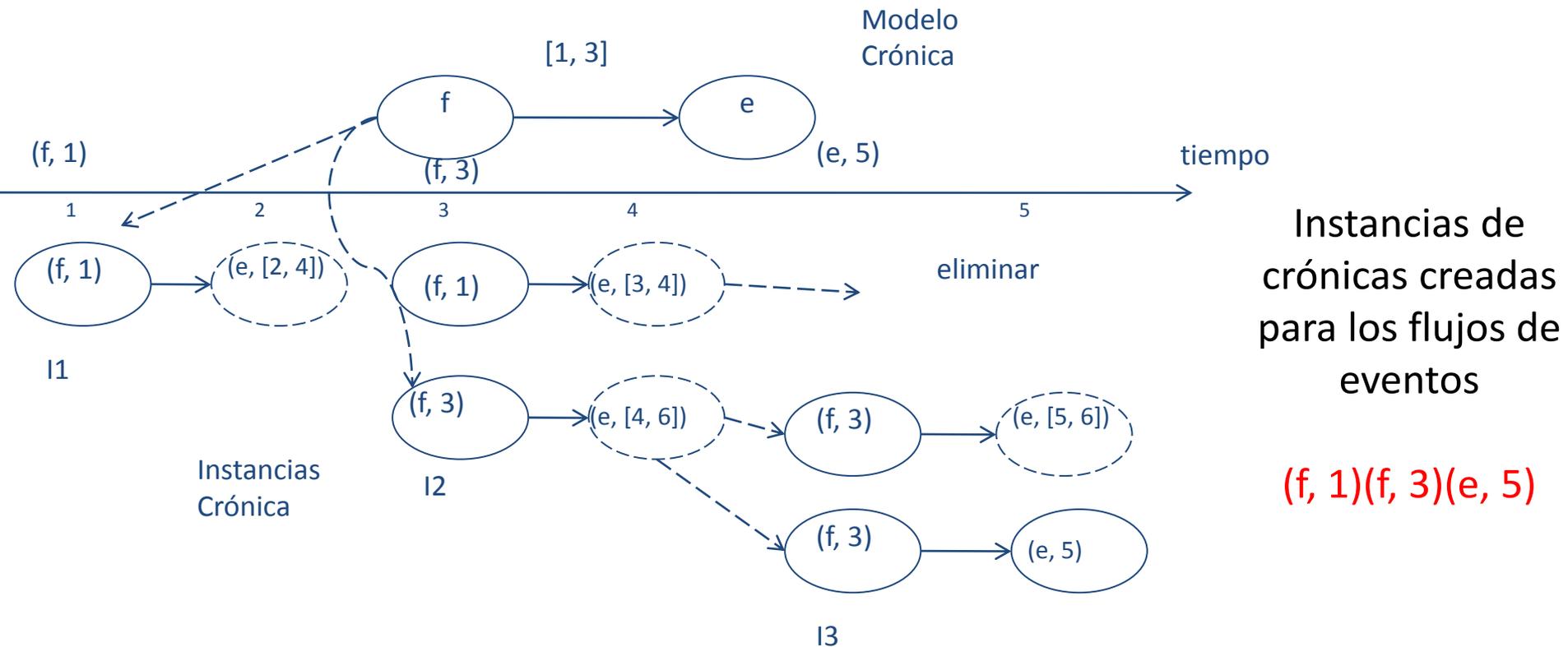


Las secuencias S-1 y S-2 son 2 instancias de la crónica C



Reconocimiento en las Crónicas

Algoritmo de Reconocimiento de (Dousson et al., 1994), (Dousson et al., 2007).



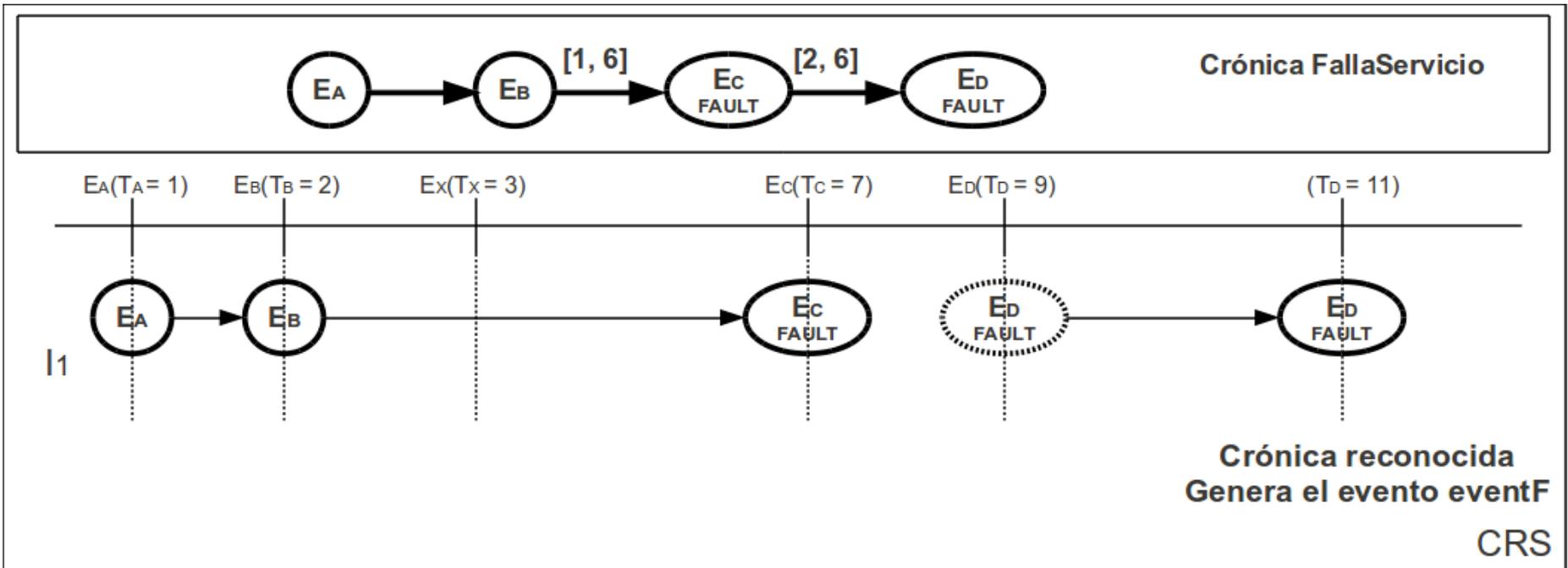
Reconocimiento en las Crónicas

- **Etapas del Reconocimiento:**
 - Evolución de la instancia parcial de HLE.
 - Reconocimiento predictivo
- **Gestión de instancias parciales de HLE**
 - Con el fin de gestionar todas las instancias parciales de HLE, CRS los guarda en árboles, uno para cada HLE.
 - Cada ocurrencia de evento y cada tic tac del reloj atraviesan esos árboles con el fin de matar algunos HLE o extender algunos HLE.
- El rendimiento del CRS dependen directamente del número de casos parciales de HLE
cada evento (kn) con k el número de instancias, n el tamaño de los modelos.

Reconocimiento en las Crónicas

```
chronicle FallaServicio() {
    //tiempos que caracterizan la crónica
        timepoint tA, tB, tC, tD0, tD1;
    //Eventos considerados
        event(EA, tA);
        event (EB, tB);
    //las aserciones (El contexto de la crónica)
        event (EC:(NoFault, Fault), tC);
        hold (ED: Fault, (tD0, tD1));
    //- Las restricciones entre los instantes
        tA < tB;
        (tC - tB) in [1, 6];
        (tD0 - tC) in [2, 6];
    when recognized {
        //Acciones a tomar cuando el reconocimiento se ha activado
            generate(eventF)";
    }
}
```

Reconocimiento en las Crónicas



Reconocimiento en las Crónicas

Enfoque de (Carrault et al., 1999), (Quiniou et al., 2001), (Carrault et al., 2003)

- La clase siguiente describe el concepto bigeminy usando eventos QRS:

```
class(bigeminy) :-  
    qrs(R0, normal, _),  
    qrs(R1, abnormal, R0),  
    rr(R1, R0, short).
```

- La crónica asociada es:

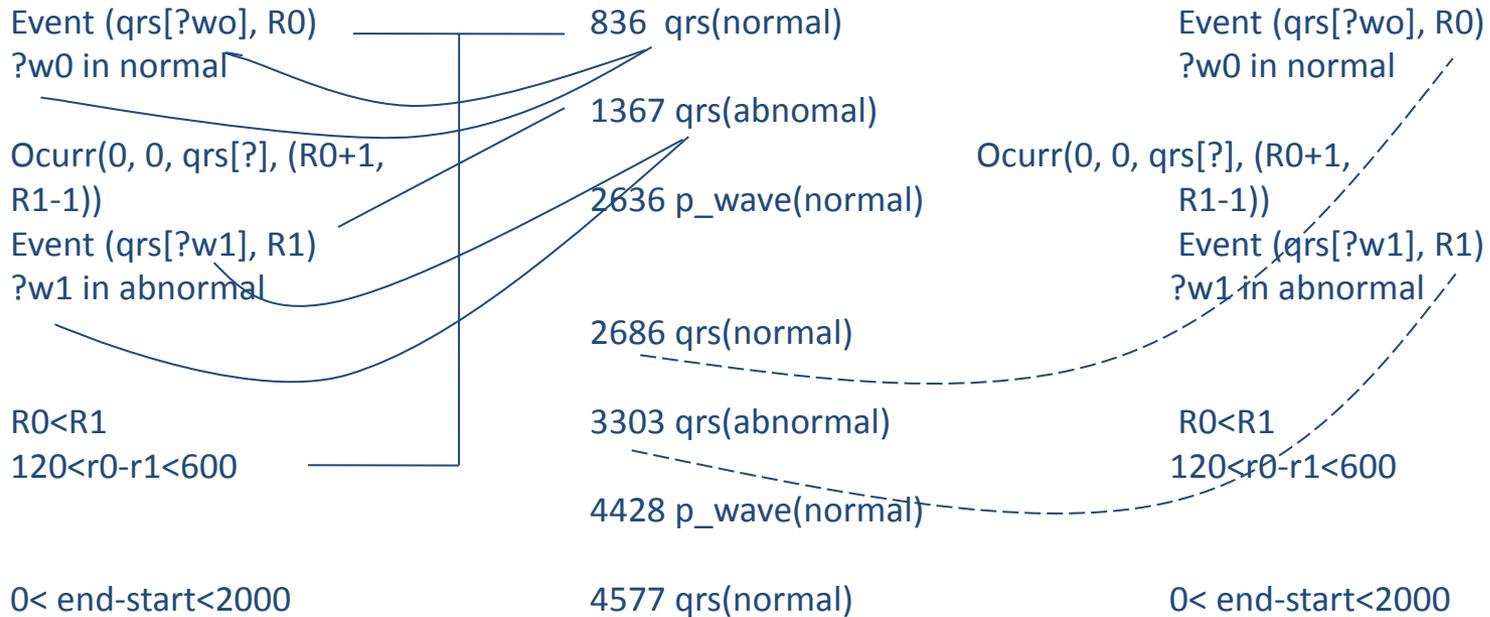
```
Chronicles bigeminy(){  
    Event (qrs[?w0,R0]; // qrs(R0, normal, -),  
    ?w0 in {normal}  
    Ocurrs(0, 0, qrs [?], (R0+1, R1-1));  
    Event(qrs[?w1],R1); // qrs(R1, abnormal, R0),  
    ?w1 in {abnormal};  
    R0<R1; // rr1(R0, R1, short),  
    120 < R1-R0 < 600;  
    0 < end-start < 2000}
```

Reconocimiento en las Crónicas

Una instancia de la crónica

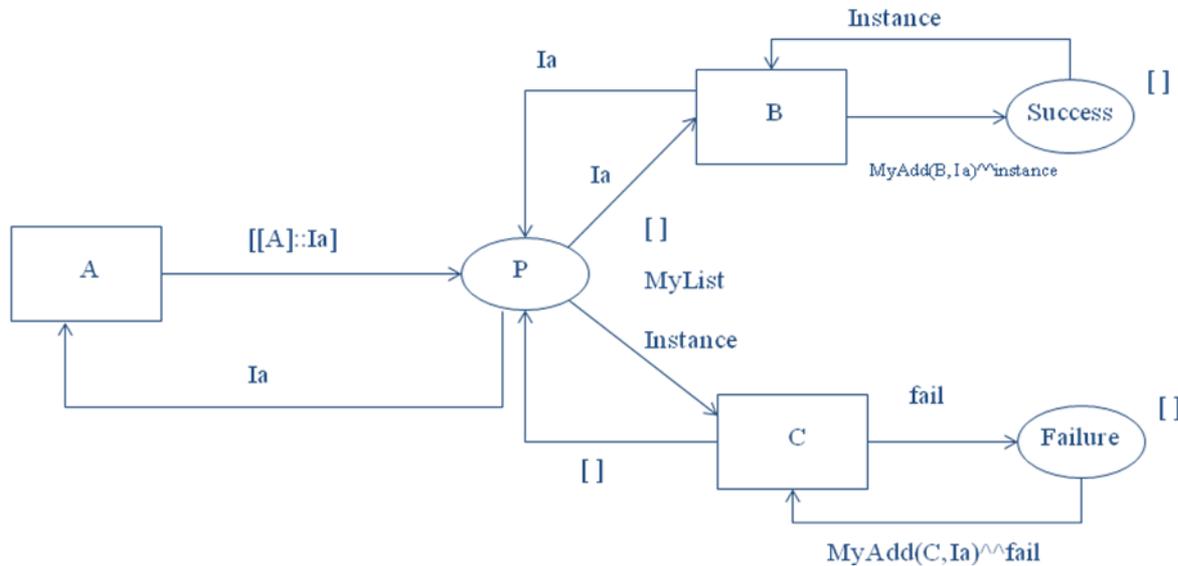
Secuencia de Eventos desde una señal ECG

Otra instancia de la crónica

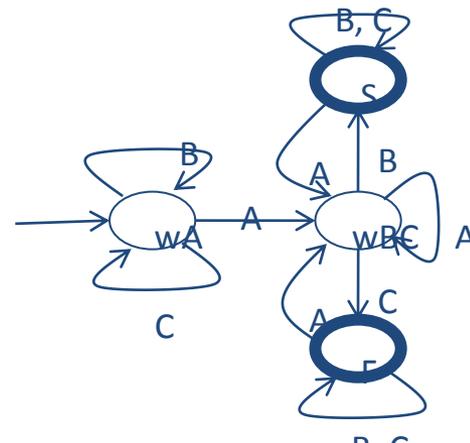


Reconocimiento en las Crónicas

En (Bertrand et al., 2007, 2008), 2009) proponen un procedimiento para reconocer crónicas basada en redes de petri coloreadas.



Enfoque de reconocimiento de crónica basado en autómatas de estado finito (Bertrand, 2007).



Reconocimiento en las Crónicas Distribuidas (Aguilar et al., 2013)

- El **reconocimiento** de una crónica distribuida debe ser **completamente distribuido**.
- Para ello, se coloca **un sistema de reconocimiento en cada componente** del sistema distribuido, el cual será responsable por **reconocer las subcrónicas en ese sitio**.
- **Los BE** le permiten a los sistemas de reconocimiento local **inferir información desde sus vecinos**, y usarla para reconocer las subcrónicas locales, generar eventos a sus vecinos, y en general, esparcir lo inferido.
- El BE la da a cada sitio una **global visión del sistema**

Reconocimiento en las Crónicas Distribuidas (Aguilar et al., 2013)

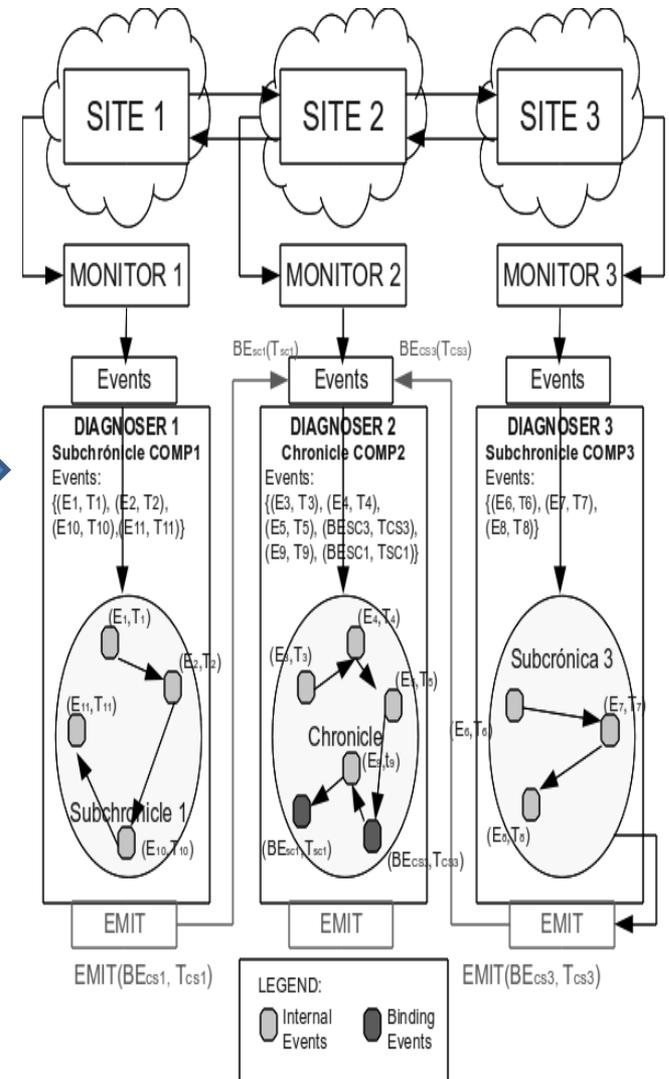
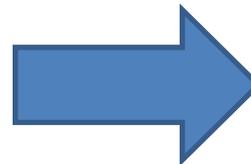
Definición como una crónica C se puede descomponer en n Subcrónicas (SC), el reconocimiento de una crónica puede ser realizada por una de sus subcrónicas SC_i , reconociendo sus eventos (Eac_i, Tac_i) , uniéndolo con el reconocimiento parcial de las otras subcrónicas $(SC_j \forall j=1, n \mid j \neq i)$ de los otros eventos (para ello, las otras subcrónicas deben enviarle un mensaje para informarle el reconocimiento de sus eventos).

Así, la subcrónica SC_i reconoce la crónica $C(E, T)$:

$$C(E, T) = \{(Eac_i, Tac_i), \text{UNION}_{j=1, n \mid j \neq i}(SC_j(Eac_j, Tac_j))\}$$

Reconocimiento en las Crónicas Distribuidas (Aguilar et al., 2013)

<p>Chronicle Subchronicle 1 { Events{ event(E_1, T_1), event(E_2, T_2), event(E_{10}, T_{10}), event(E_{11}, T_{11}) } Constrains{ $T_2 - T_1 \leq C_1$ $T_{10} - T_2 \leq C_2$ $T_{11} - T_{10} \leq C_3$ } When recognized{ Emit event(BE_{SC1}, T_{SC1}) to Diagnoser 2 } }</p>	<p>Chronicle Subchronicle 2 { Events{ event(E_3, T_3), event(E_4, T_4), event(E_5, T_5), event(BE_{SC3}, T_{SC3}), event(E_9, T_9), event(BE_{SC1}, T_{SC1}) } Constrains{ $T_4 - T_3 \leq C_4$ $T_5 - T_4 \leq C_5$ $T_{SC3} - T_5 \leq C_6$ $T_{SC1} - T_9 \leq C_8$ } When recognized{ Create log(Fault 1) } }</p>	<p>Chronicle Subchronicle 3 { Events{ event(E_6, T_6), event(E_7, T_7), event(E_8, T_8) } Constrains{ $T_7 - T_6 \leq C_9$ $T_8 - T_7 \leq C_{10}$ } When recognized{ Emit event(E_{SC3}, T_{SC3}) to Diagnoser 2 } }</p>
---	--	--



Reconocimiento en las Crónicas Distribuidas (Aguilar et al., 2013)

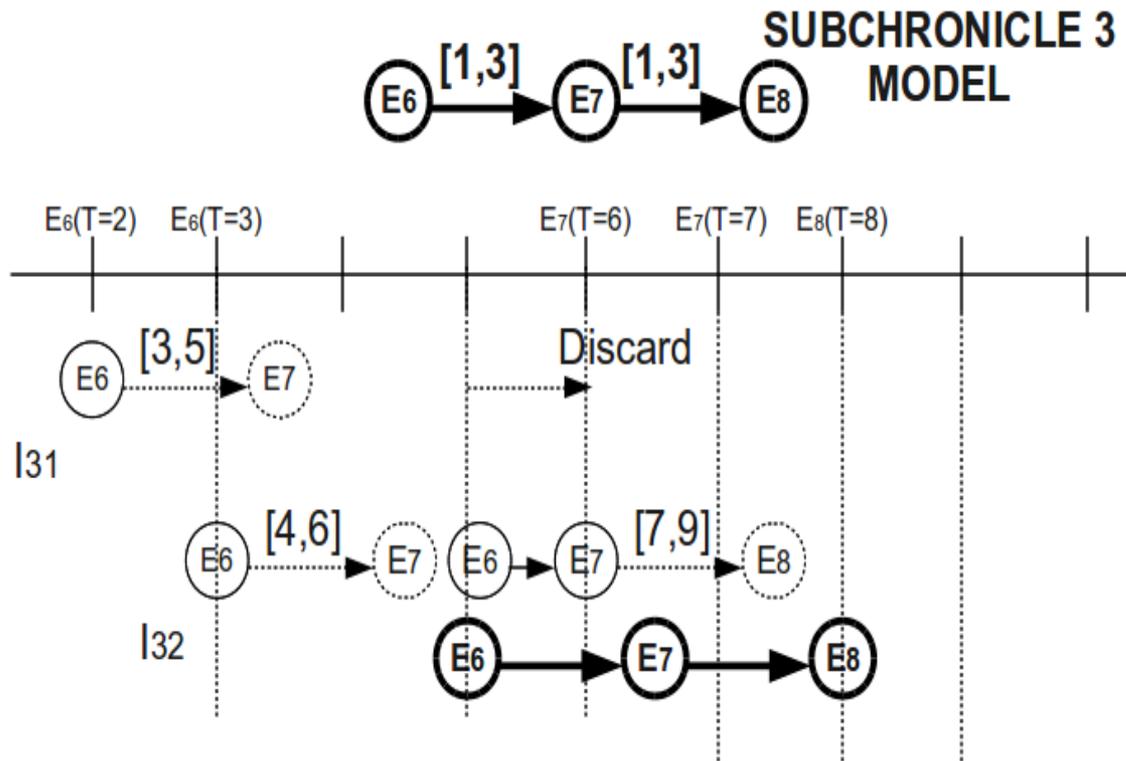
El programa CRS local

1. FOREACH event received DO
2. diagnoser.addEvents(chronicle c, event)
3. IF chronicle c is recognized THEN
- 4 DO c.executeAction()

diagnoser.addEvents (chronicle c, event)

- 1 FOR EACH chronicle c where event is the first event DO
 - 1.1 instances = c.instances();
- 2 FOR EACH current instances DO
 - 2.1 IF event match instances and temporal constraints are not violated THEN
 - 2.1.1 instances.addEvent(event)
 - 2.2 IF event match instances and temporal constraints are violated THEN
 - 2 .2.1 instances.discard()
 - 2.3 IF temporal constraints are violated THEN
 - 2.3.1 instances.discard()

Reconocimiento en las Crónicas Distribuidas





Aprendizaje en Crónicas

Jose Aguilar

Aprendizaje en Crónicas

- En general, definiremos **aprendizaje** como la capacidad para automáticamente adquirir conocimiento
- Los beneficios del aprendizaje en las crónicas son:
 - Una más fácil **adquisición de conocimiento**,
 - **Crónicas discriminantes** son aprendidas (eso asegura un eficiente reconocimiento de crónicas),
 - Fácil **mantenimiento de la Base de Crónicas**.
- En IA, las técnicas de aprendizaje se pueden clasificar en: **aprendizaje *supervizado***, **aprendizaje *no supervizado***, **aprendizaje por *reforzamiento***,

Para el paradigma de crónicas esa clasificación es interesante!!

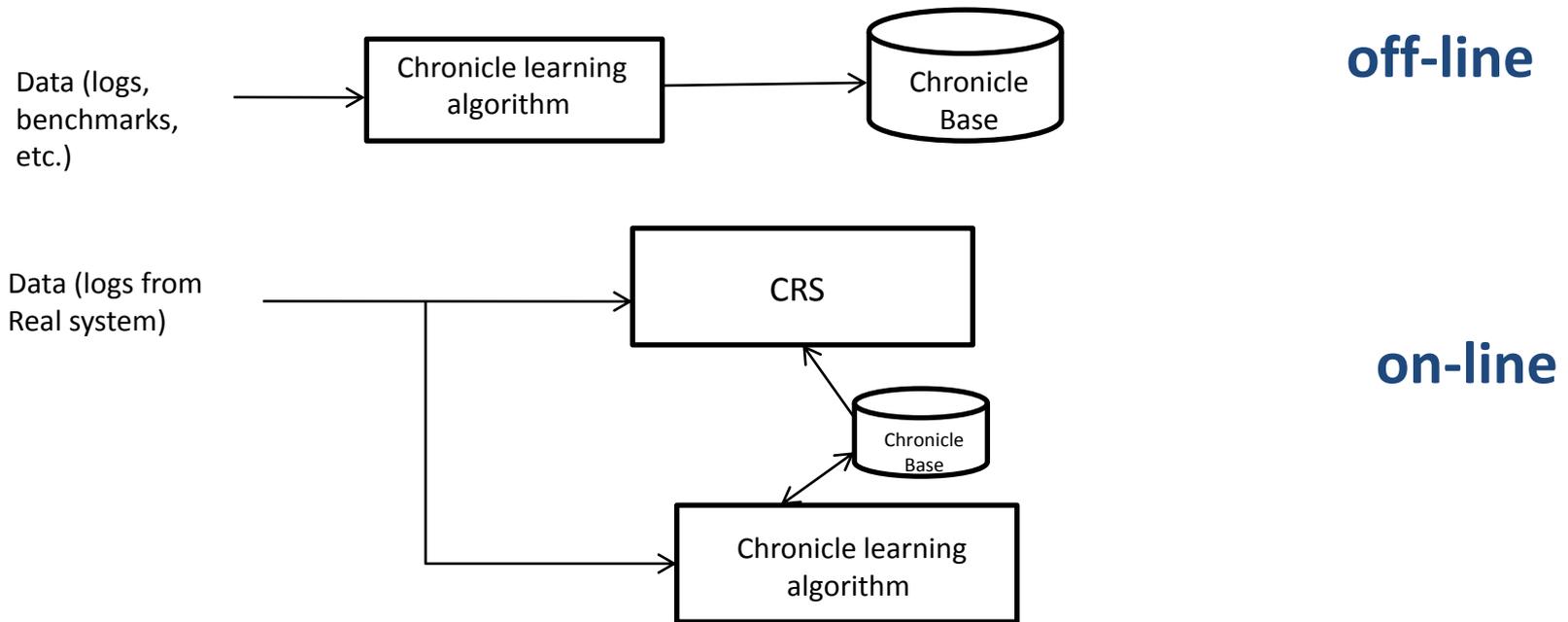
Aprendizaje en Crónicas

Problema clásico de aprendizaje en crónicas consiste en tener un conjunto de hechos sobre eventos y el mundo,

¿Como descubrir relaciones causales con esa información?

El resultado es la construcción de crónicas que describen esa situación

Aprendizaje en Crónicas



Aprendizaje en Crónicas

Dos tipos

- **Caso en que el esqueleto de las crónicas son definidos a mano (eventos y sus relaciones) , y el aprendizaje se limita a aprender las restricciones temporales entre los eventos.** Las crónicas son así aprendidas desde ejemplo dados como secuencias de LLE. Pueden ser:
- **Aquellas que identifican el esqueleto de las crónicas como las restricciones temporales :** Las crónicas son así aprendidas desde ejemplo directamente dados como series temporales.

Mecanismos de extracción de información Temporal: consideran datos temporales, específicamente series de tiempo multivariables muestreados desde señales que describen el proceso bajo estudio, para construir patrones temporales

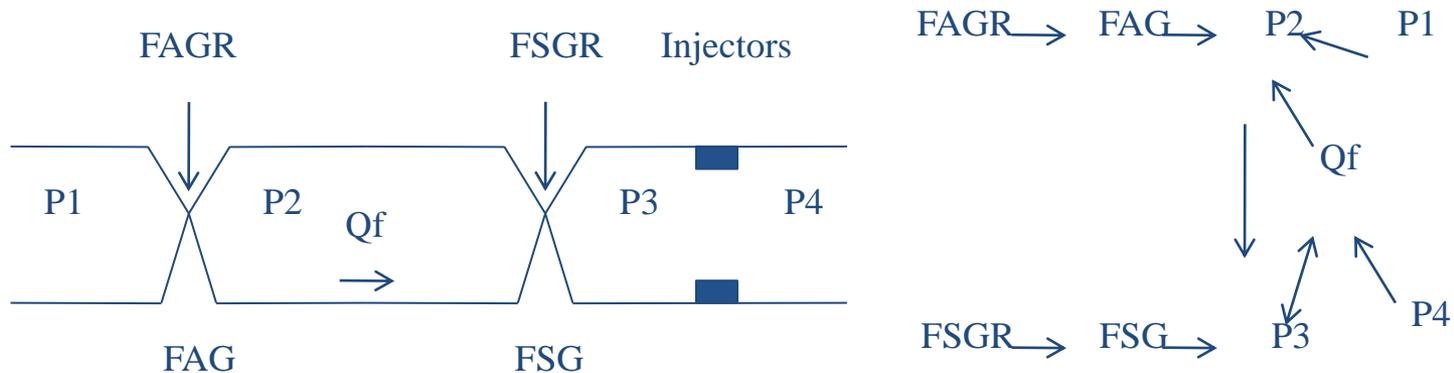
Aprendizaje en Crónicas

Caso en que el esqueleto de las crónicas son definidos a mano (eventos y sus relaciones) , y el aprendizaje se limita a aprender las restricciones temporales entre los eventos.

- Basadas en modelos cualitativos.
- Basadas en datos históricos
- Basadas en técnicas de Minería de Datos sobre archivos log para descubrir patrones (clases).
- Basado en técnicas de lógica inductiva (ellas requieren definir las crónicas como sentencias en lógica de primer orden).

Basadas en modelos cualitativos

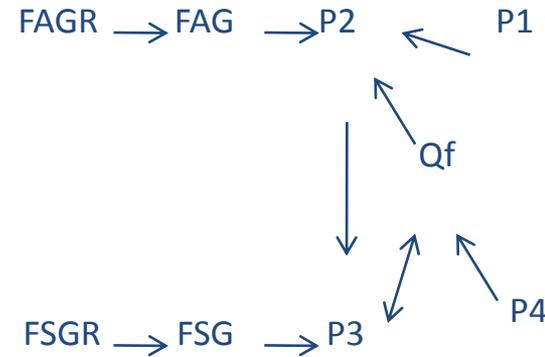
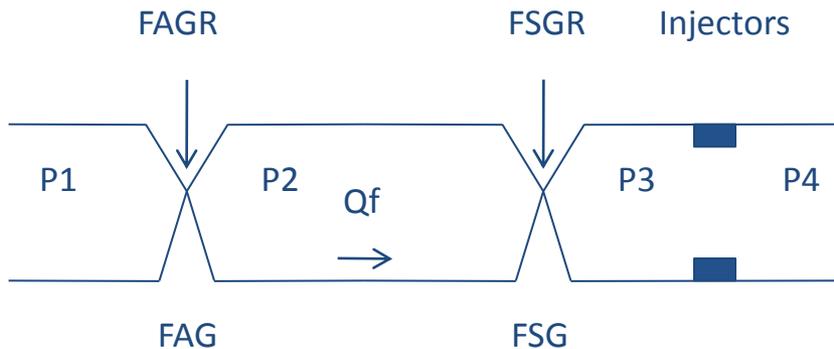
- Modelo cualitativo definido como un grafo causal de los parámetros/variables del sistema real.



- Los eventos son definidos como los instantes de tiempo donde cambian los valores de las variables que describen el sistema
- Se escribe el modelo de crónica que representa la relación entre ellos. Se desarrollan dos crónicas mutuamente excluyentes entre ellas.

Basadas en modelos cualitativos

- Modelo



- Cronicas.

```
Chronicle NormalCasualLink ()
{
    timepoint ta, tb;
    Event (FAGR: (steady, increasing), ta);
    Event (FAG: (steady, increasing), tb);
    D(tb-ta)=(MinAB, maxAB);
    When recognized
        Hold (FAG_link: normal, (ta, tb));
}
```

```
Chronicle AbnormalCasualLink ()
{
    timepoint ta, tb;
    Event (FAGR: (steady, increasing), ta);
    Hold (FAG: (steady, (ta, tb)));
    D(tb-ta)> maxAB;
    When recognized
        Hold (FAG_link: abnormal, (ta, tb));
}
```

Basadas en modelos cualitativos

- Se definen las relaciones temporales entre los eventos por algún método. Por ejemplo, por simulación, predicción, etc. desde un estado inicial
- Los expertos determinan si los umbrales para determinar los cambios de valores de las variables son los adecuados, así como válida los eventos definidos

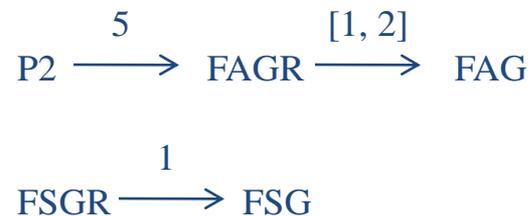
Basadas en datos históricos

- Los datos históricos se traducen en una lista de eventos con fecha.
- Se extraen secuencia de eventos característicos de las fallas estudiadas (se establece la estructura del grafo causal de la falla).
 - para el ejemplo anterior; suponer siguiente histórico de las fallas de las válvulas FAG, FSG (eventos y fecha de ocurrencia)
 1. P2:0 FAGR: 5, FAG 6, FSGR: 7, FSG :7
 2. P2:0 FSGR: 3, FSG 4, FAGR: 5, FAG: 6
 3. FSGR: 0, FSG: 1, P2: 2, FAGR: 7, FAG: 9

Las sub-palabras comunes: {(P2, FAGR, FAG), (FSGR, FSG)}

Basadas en datos históricos

- Se determinan las restricciones temporales:
 - Se emparejan los eventos de los datos históricos con las subpalabras antes determinadas.
 - Se usa el proceso de reconocimiento de crónicas donde las subpalabras son las crónicas y la data histórica el flujo de entrada.
- Para el ejemplo previo:



Basadas en datos históricos

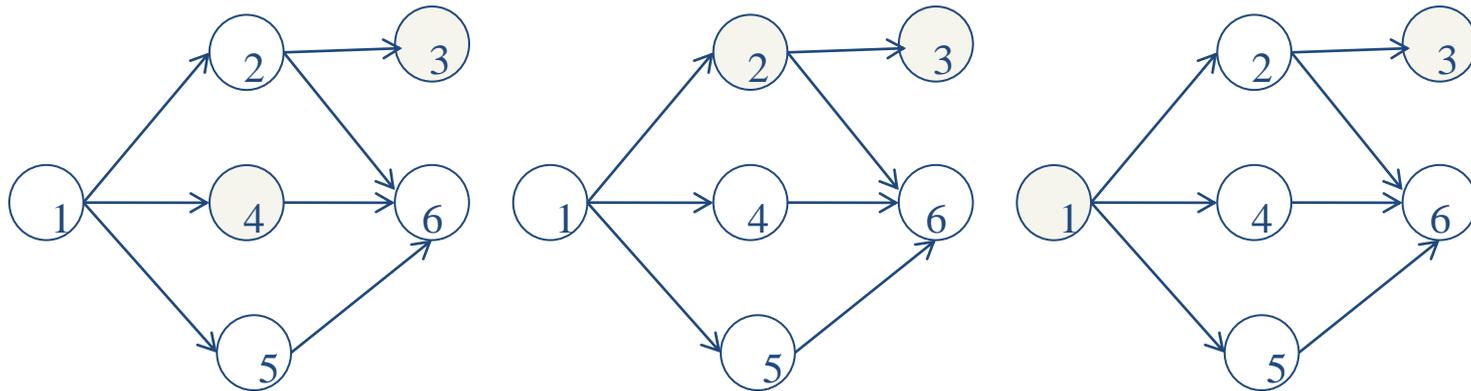
- Se analiza con contra ejemplos para **especializar** las crónicas.
 - Para cada sub-crónica se determina si un contra-ejemplo puede reconocerla.
 - Entonces, con las sub-crónicas no reconocidas se establece la combinación mínima de ellas para encontrar la crónica de la situación (por ejemplo, una falla).
- Por ejemplo, si tenemos 6 contra-ejemplos y 4 sub-crónicas, la tabla siguiente muestra si son reconocidas (*) o no (-):

	1	2	3	4	5	6
SC1	*	*	-	-	-	-
SC2	-	-	-	*	-	-
SC3	-	-	-	-	-	-
SC4	*	*	*	*	*	*
SC5	-	*	-	-	-	-

Según esa tabla, hay 3 posibles crónicas {SC1 \wedge SC2, SC3, SC2 \wedge SC5}. No se sabe cual es mejor, y la unión de ellas sería lo mejor

Basadas en datos históricos

- Ahora **se generaliza** la crónica. Se puede hacer al tenerse contra-ejemplos muy específicos y se requiere que la crónica no los acepte. Para eso, se deben borrar partes de las secuencias de eventos que la componen.



Se eliminar los nodos presentes en todas las reacciones (nodos 5 y 6).

El color gris representa los eventos de la crónica de falla que no son reconocidos por los contra-ejemplos

Aprendizaje en Crónicas (Aguilar et al., 2011)

1. Pre-procesar
2. Extraer descriptores
3. Especificar patrones primitivos (clases)
4. Identificar Sucesiones
5. Detectar eventos
6. Extraer secuencias
7. Especificar patrones temporales (crónicas)
8. Actualizar Base de Crónicas

Aprendizaje en Crónicas (Aguilar et al., 2011)

Clasificación de Descriptores del sistema (SALSA)

Ejemplos Descriptores = Individuo
« retraso, tasa de envió, %perdida »

Definir los cambios de
estado en el sistema

Metodología de
agrupamiento difuso
LAMDA (Learning
Algorithm for
Multivariate Data
Analysis) base de la
herramienta SALSA.

Aprendizaje en Crónicas (Aguilar et al., 2011)

Conceptos de base:

✚ individuo: Representa un vector (retardo, *tasa de envío*, etc.) calculado al instante t , definido como $indiv_i$:

\forall la clase $C_j \exists \mu_{ij} \geq 0$ Grado de pertenencia de $indiv_i$ a C_j

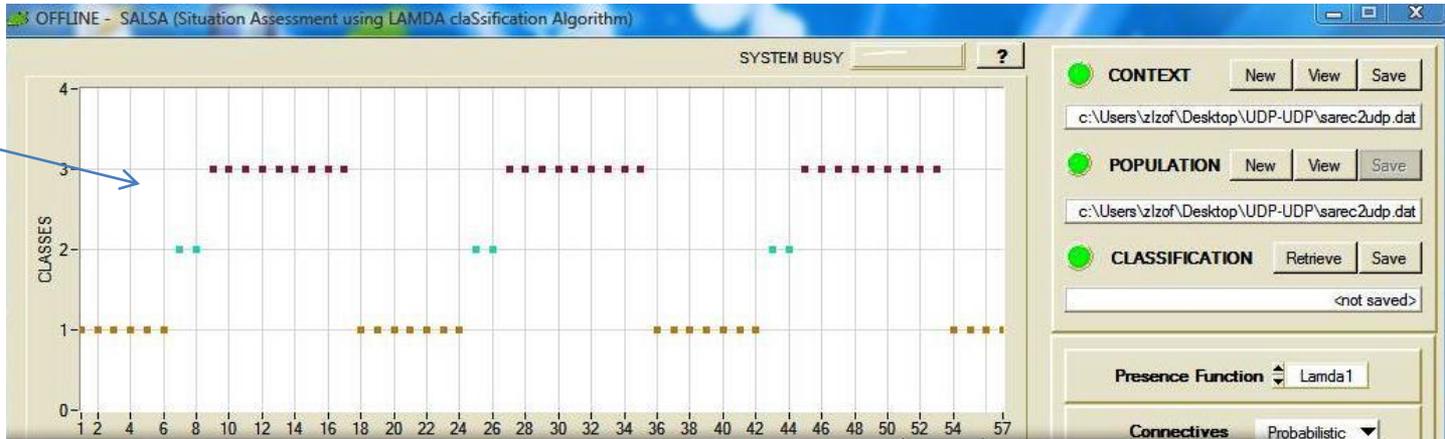
Determina el grado de pertenencia de un individuo a una clase, considerando el aporte de cada atributo (en los descriptores), eso se llama MDA (marginal adequacy degree)

✚ clase: resultado de SALSA, definido C_j :

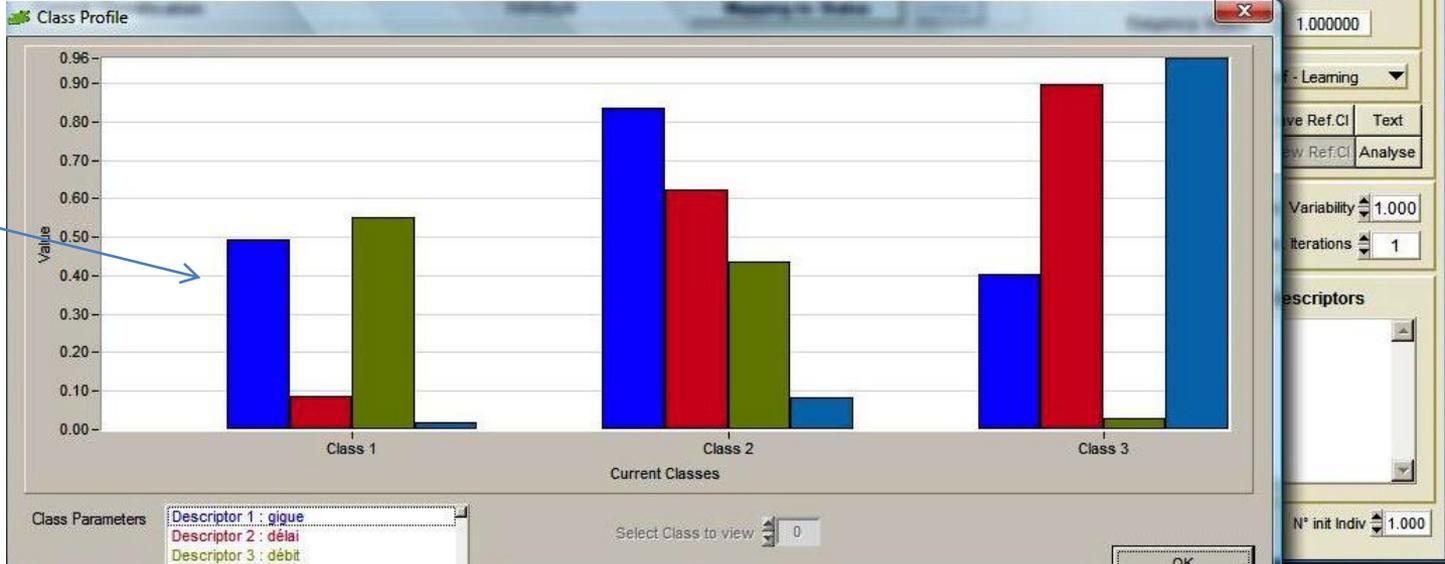
- $\mu_{max j}$ Grado de pertenencia máximo
- $\mu_{min j}$ Grado de pertenencia mínimo
- $\#indiv_j$ Número de individuos

Aprendizaje en Crónicas (Aguilar et al., 2011)

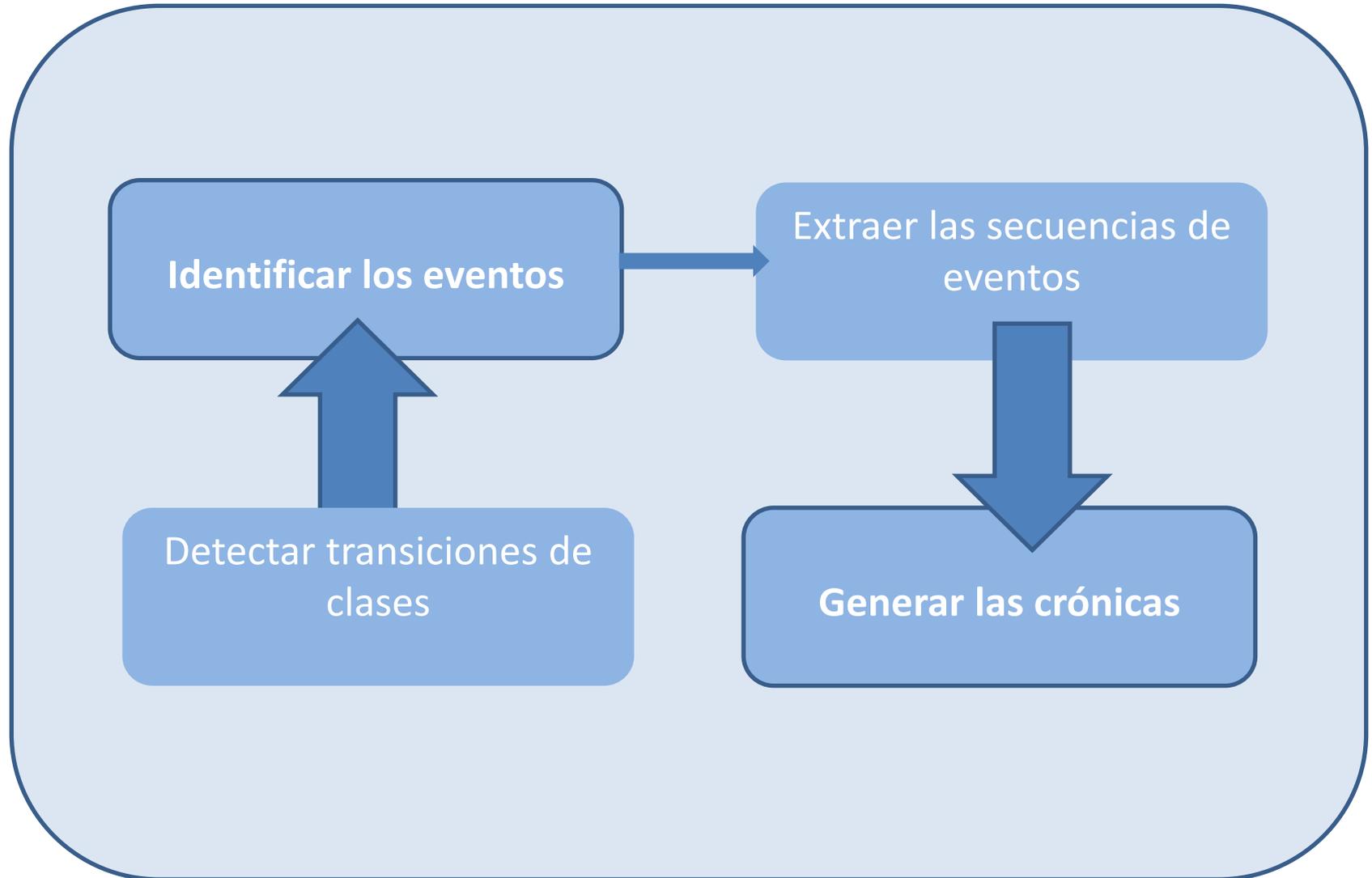
Individuos



Perfil clases



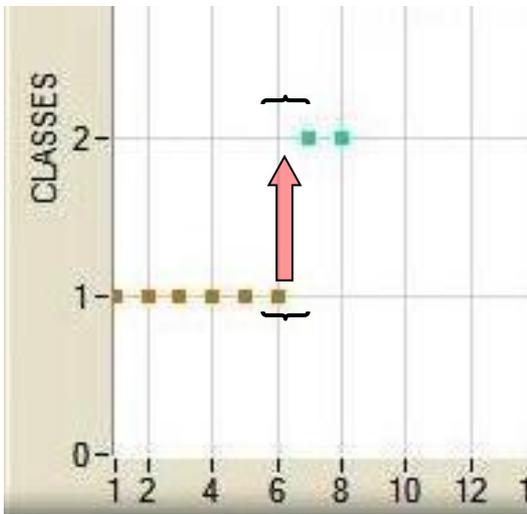
Aprendizaje en Crónicas (Aguilar et al., 2011)



Aprendizaje en Crónicas (Aguilar et al., 2011)

Detectar transiciones de clases: **cambios de estado**

- Dos individuos
- Cambio Instantáneo

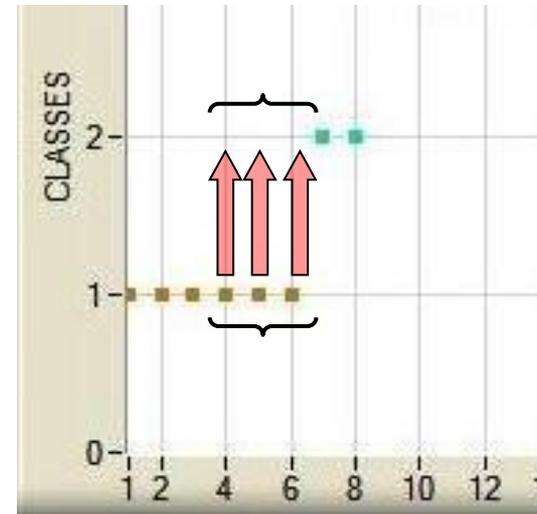
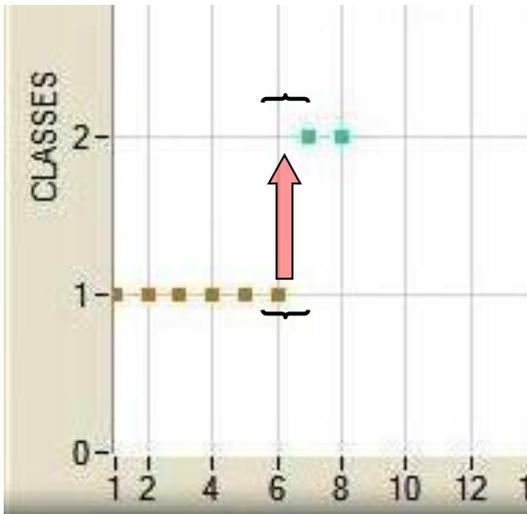


Aprendizaje en Crónicas (Aguilar et al., 2011)

Detectar transiciones de clases: **cambios de estado**

- Dos individuos
- Cambio Instantáneo

✚ Objetivo: alargar zona de transición



✚ Necesidades:

- Considerar cambios de estados progresivos
- Integrar las ventajas del aspecto temporel

Aprendizaje en Crónicas (Aguilar et al., 2011)

Detectar transiciones de clases :

✚ Umbral basado en la función de pertenencia

$$Seuil_{\alpha} = \mu_{max_{\alpha}} - \left(\frac{\mu_{max_{\alpha}} - \mu_{min_{\alpha}}}{\#indiv_{\alpha}} \right)$$

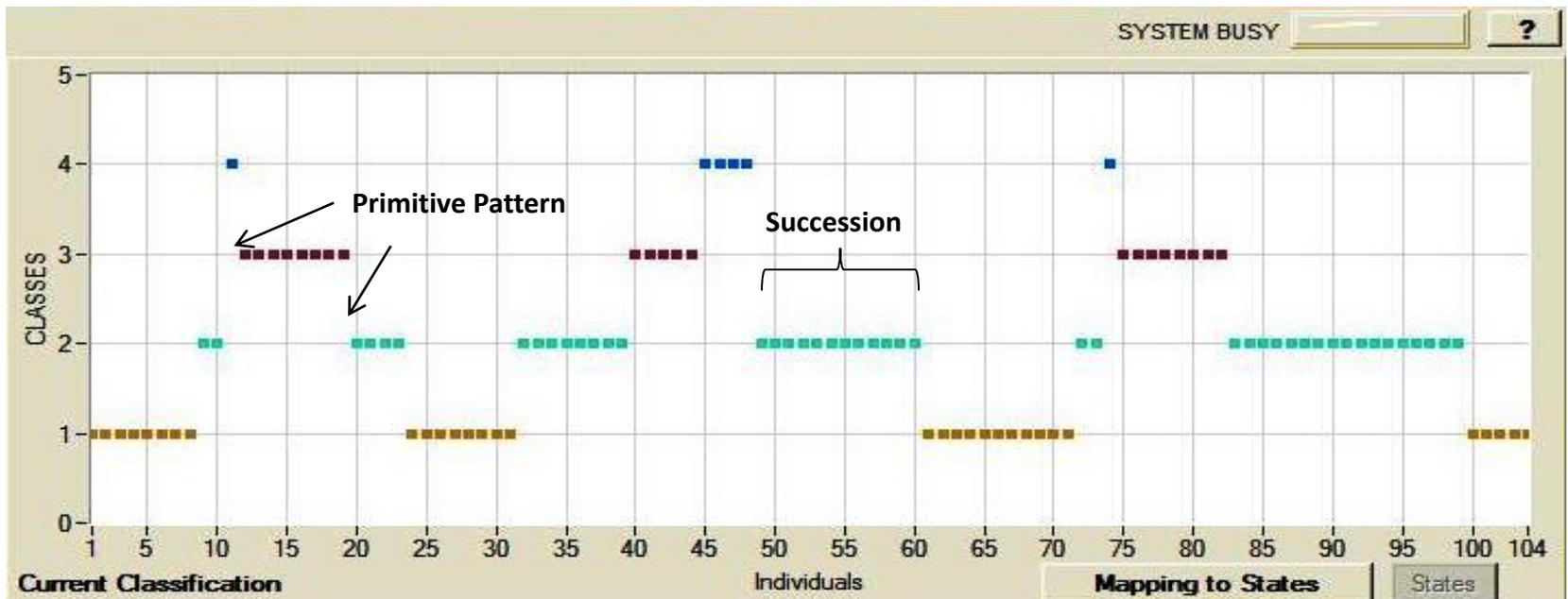
Sea C_{α} , $\forall indiv_i \in C_{\alpha}$

Si $\mu_{i_{\alpha}} \geq Seuil_{\alpha}$

entonces $indiv_i$ es una transición

Aprendizaje en Crónicas (Aguilar et al., 2011)

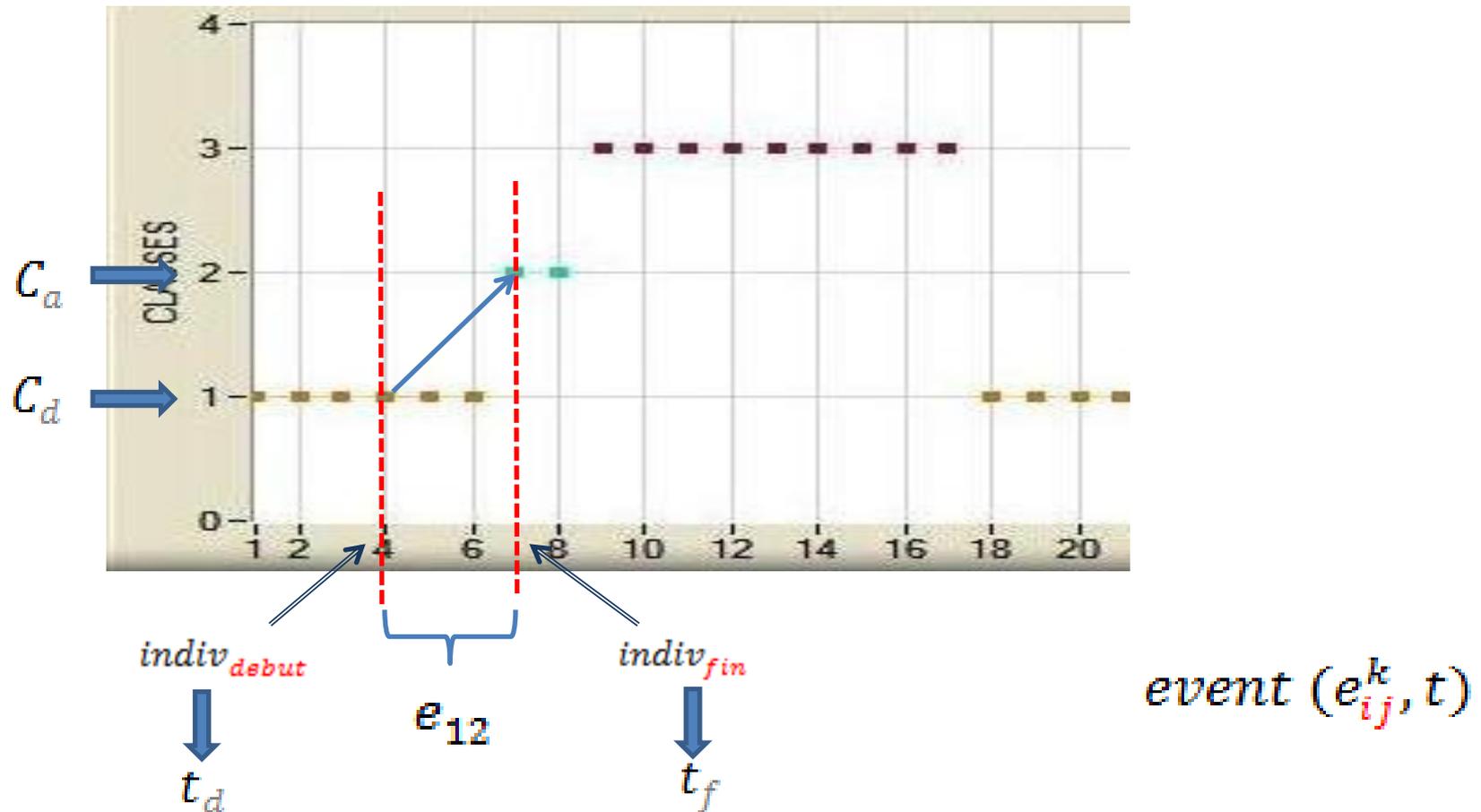
Patrones primitivos y sucesiones



Aprendizaje en Crónicas (Aguilar et al., 2011)

Identificación de eventos:

- ✚ Un evento es caracterizado por el individuo de inicio y el individuo final de una transición



Aprendizaje en Crónicas (Aguilar et al., 2011)

Noción de Similaridad:

Sea : e_{xy}

Si Existe : e_{xn} e_{ny}

Entonces secuencia : e_{xn} e_{ny} es similar a e_{xy}

Aprendizaje en Crónicas (Aguilar et al., 2011)

Extracción de secuencias de eventos:

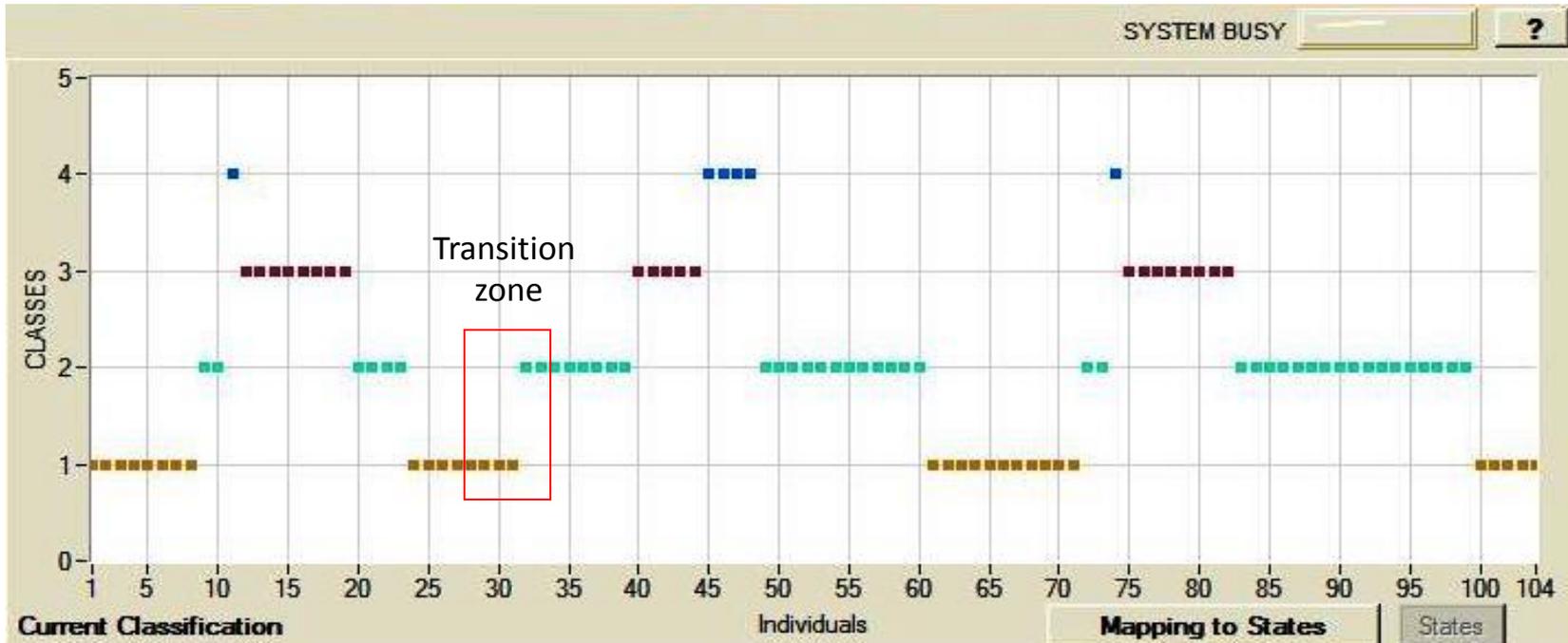
✚ usar noción de similaridad entre eventos

$$Seq_1 = \{ e_{12}, e_{21}, e_{13}, e_{32}, e_{21}, e_{13}, e_{31}, e_{12}, e_{24}, e_{41} \}$$

$$Seq_2 = \{ e_{13}, e_{32}, e_{23}, e_{31}, e_{13}, e_{31}, e_{12}, e_{21} \}$$

$$Seq_3 = \{ e_{13}, e_{31}, e_{12}, e_{24}, e_{42}, e_{21}, e_{13}, e_{31}, e_{12}, e_{24}, e_{41} \}$$

Detectar eventos y Extraer secuencias

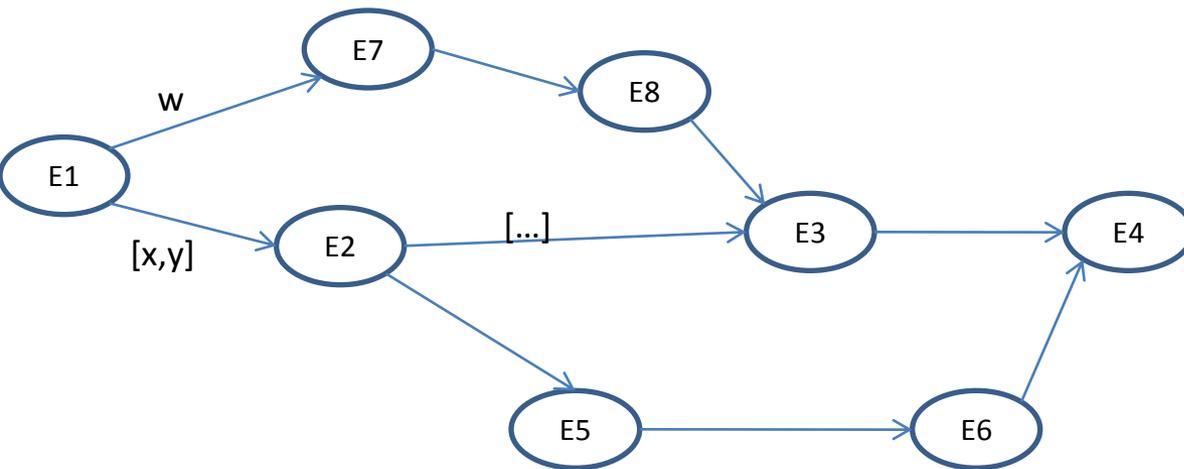


Secuencia 1 : E1, E2, E3, E4

Secuencia 2: E1, E2, E5, E6, E4

Secuencia 3: E1, E7, E8, E3, E4

Especificar patrones temporales (crónicas)



1. 4 patrones conexión:

- Secuencia
- Fork
- Join
- OR

2. Aspectos dinámicos: restricciones temporales

➤ secuencia 1: E1 T= 0.89
 E2 T= 1.22

restricciones temporales entre E1 y E2:

➤ secuencia 2: E1 T= 3.23 [0.33, 0.80]
 E2 T= 4.03

Especificar patrones temporales (crónicas)

Construir crónicas:

+ Objetivo:

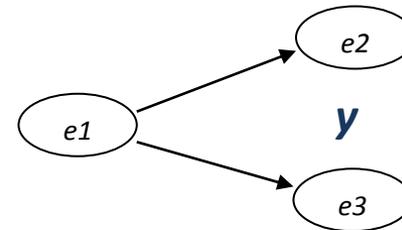
- conseguir patrón general
- deducir restricciones temporales

+ 3 etapas: buscar patrones conexión entre eventos

Buscar eventos que se repiten en todas las secuencias y guardan el mismo orden



Buscar eventos que se repiten en todas las secuencias pero en diferentes ordenes



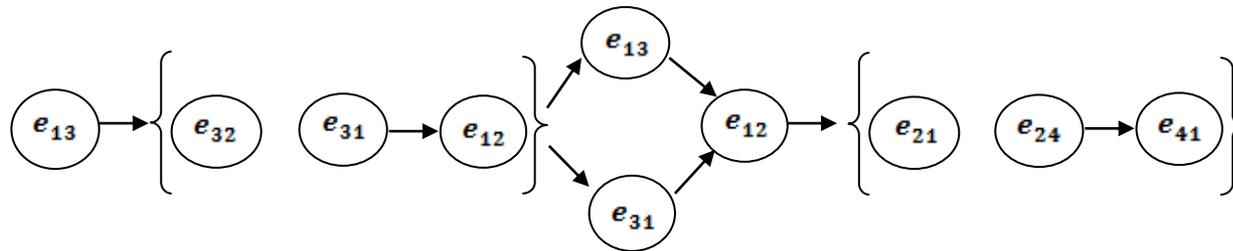
Buscar eventos similares



Especificar patrones temporales (crónicas)

Construir crónicas:

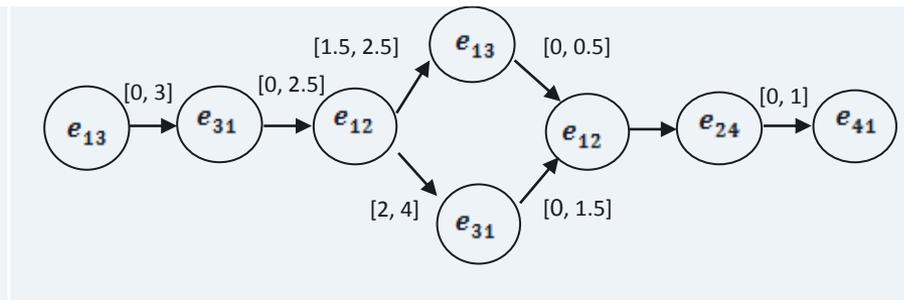
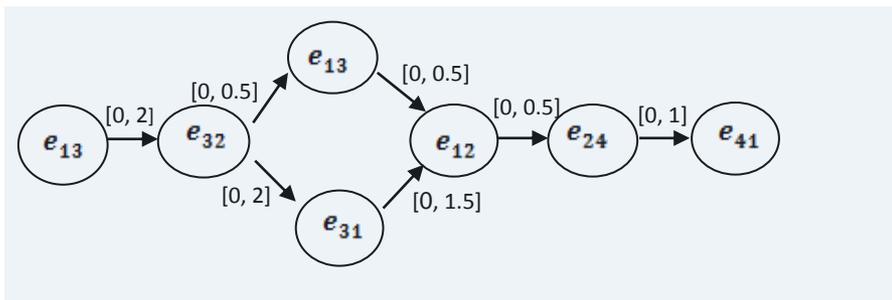
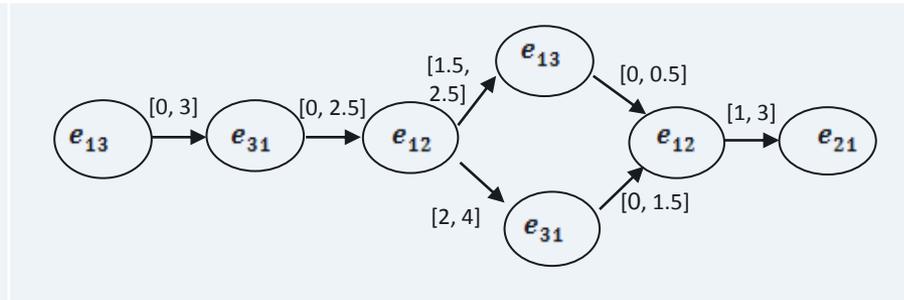
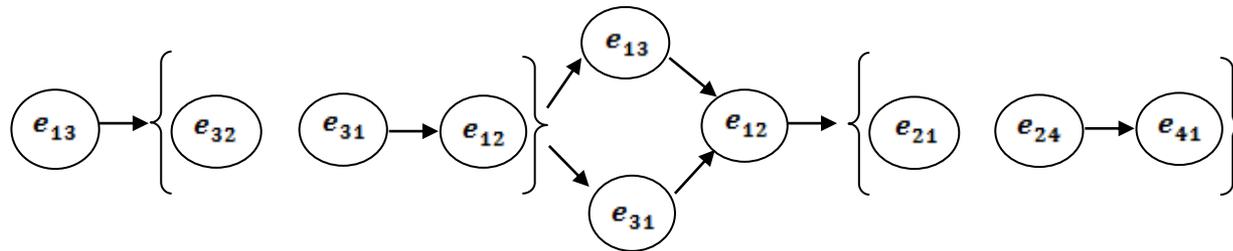
$e_{13}, (e_{23} \circ e_{31}, e_{12}), (e_{13} \text{ et } e_{31}), e_{12}, (e_{21} \circ e_{24}, e_{41})$



Especificar patrones temporales (crónicas)

Construir crónicas:

e_{13} , (e_{23} **o** e_{31} , e_{12}), (e_{13} et e_{31}), e_{12} , (e_{21} **o** e_{24} , e_{41})



Especificar patrones temporales (crónicas)

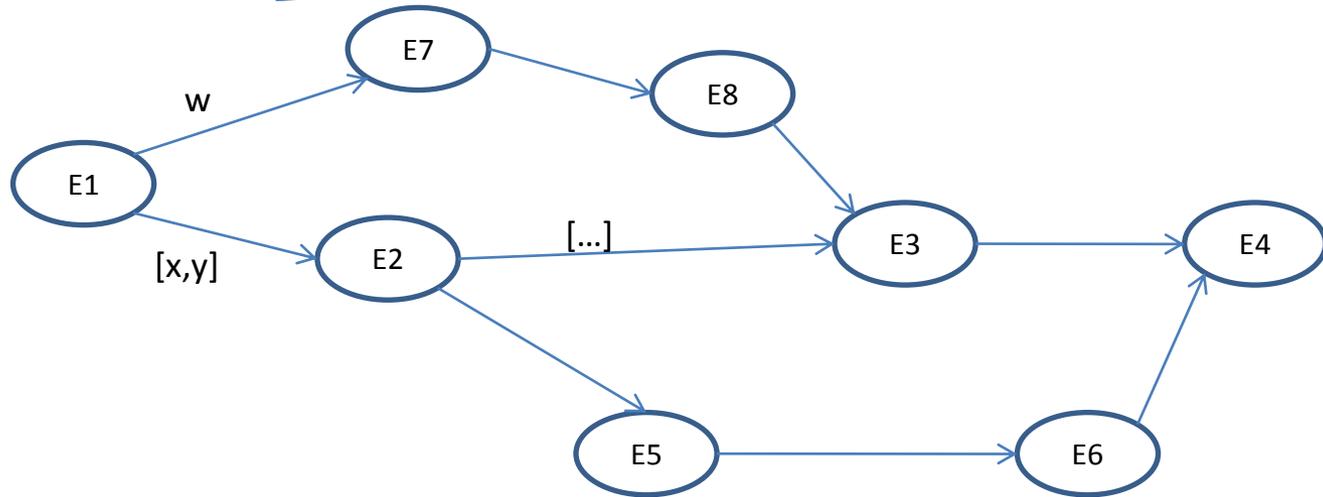
Construir crónicas:

Secuencia 1 : E1, E2, E3, E4

Secuencia 2: E1, E2, E5, E6, E4

Secuencia 3: E1, E7, E8, E3, E4

→ Crónica :



Restricción temporal entre E1 y E2:

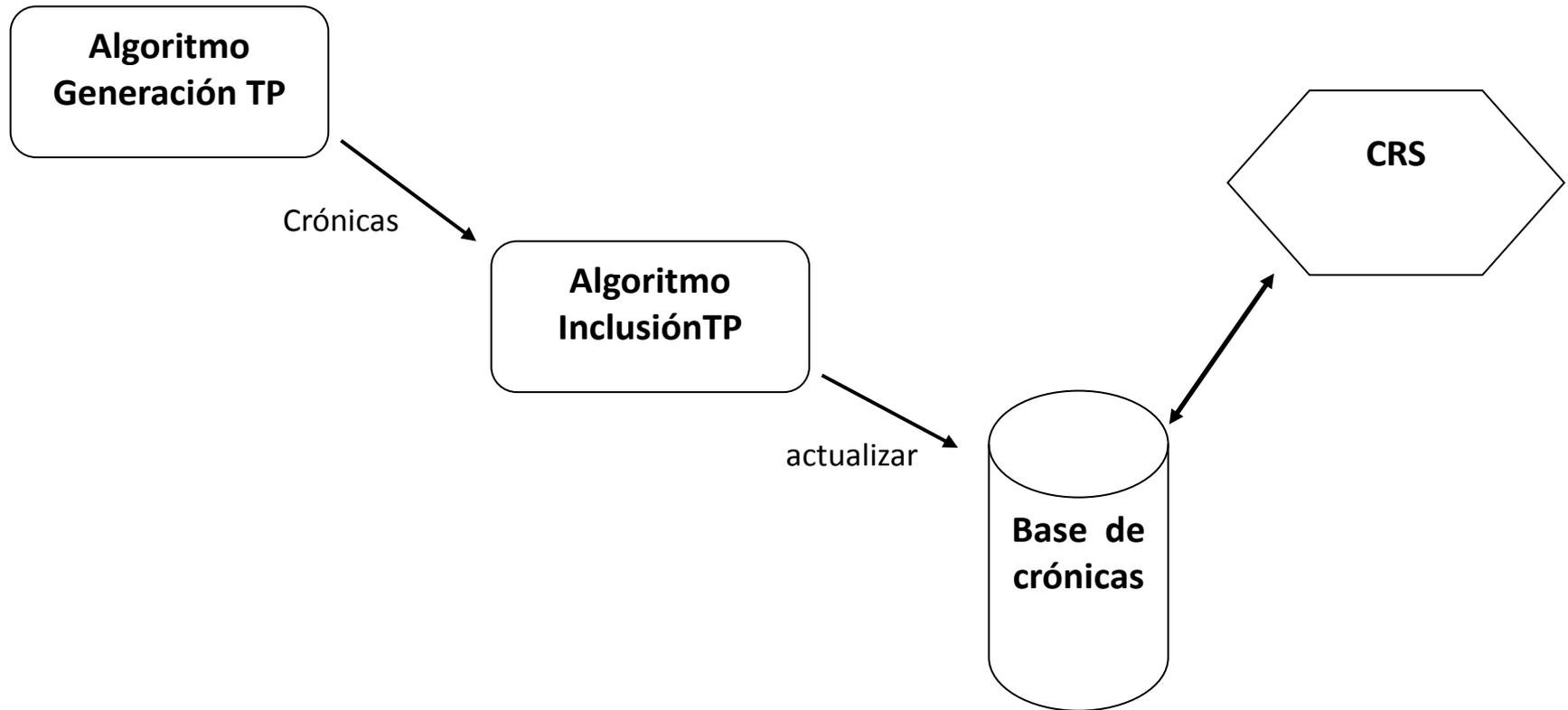
➤ 1 secuencia : E1 a t= 0.89144
E2 a t= 1.2244

X = 0.33296 s

➤ 2 secuencia : E1 a t= 3.23384
E2 a t= 4.09512

Y = 0.86128 s

Actualizar Base de Crónicas



Actualizar base de Crónicas

% Inclusion of the temporal pattern Si on the CB

For each $E' \in \text{CB}$

Calculate coverage $\text{Si}' \setminus E'$ ($\text{cov}_{E'}(\text{Si})$)

If $\text{cov}_{E'}(\text{Si})=1$

ignore

else

If $\text{cov}_{E'}(\text{Si}) < \text{Fmax}$

add Si to CB

else

if $\text{cov}_{E'}(\text{Si}) > \text{Fmax}$

update E' using Si

$$\text{cov}_{s1}(S2) = \begin{cases} 0 & \text{if } S1 \neq S2 \\ \frac{\prod_{i=1}^n \max(0, \min(u_i^1, u_i^2) - \max(l_i^1, l_i^2))}{\max(\prod_{i=1}^n u_i^1 - l_i^1, \prod_{i=1}^n u_i^2 - l_i^2)} & \text{else} \end{cases}$$

% Determination if the new chronicle covers previous chronicles stored on the CB

If Si has been added CB

For each $E' \in \text{CB}$ and

Calculate coverage $E' \setminus \text{Si}$ ($\text{cov}_{\text{Si}}(E')$)

If $\text{cov}_{\text{Si}}(E') > \text{Fmin}$

remove E' from CB

Uso de las Crónicas

Jose Aguilar

Uso de las Crónicas

- Se utilizan para **modelar actividades** a ser detectadas, porque no son deseadas o son peligrosas.
- **Una actividad se describe** mediante una combinación de ocurrencias de eventos.
- El objetivo es **identificar todas las instancias de la crónica** dentro de un flujo de eventos observados
- **La identificación de la crónica** se logra a través de la comparación entre los eventos que van ocurriendo y los eventos que describen la crónica,
- Además, puede ser de interés guardar la información que indica **cuales acontecimientos en el flujo contribuyen** al reconocimiento de la crónica, ya que puede ayudar a encontrar las causas del fenómeno observado.

IAm

Conformada por artefactos inteligentes que interactúan entre si para alcanzar objetivos del ambiente

- El sistema de comunicación debe aprender cuales artefactos interactúan en un ambiente dinámico, para responder a:
 - Como comunicarse,
 - Con quien cooperar,
 - Como delegar y coordinar tareas, etc,

de manera auto-organizada



Debe ser un proceso cognitivo distribuido

- **Sistemas de comunicación autónomos** aprenden crónicas que describen situaciones en el ambiente basada en el conocimiento sobre la plataforma de comunicación, requerimiento de usuarios, etc.

Crónicas para auto-adaptar estrategias en Sistemas de Comunicación

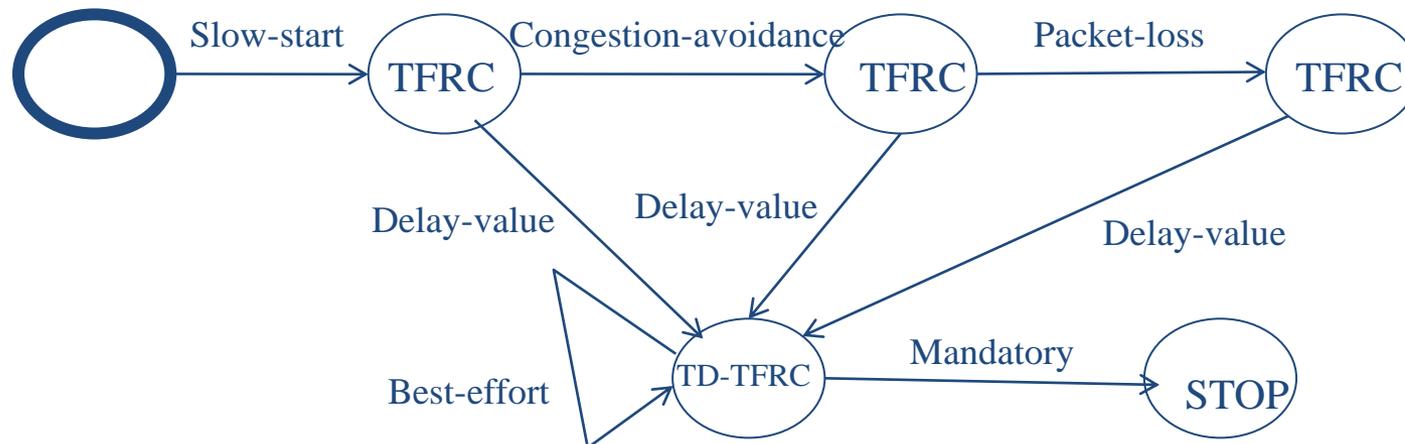
Reconocimiento de diferentes situaciones es necesario para reconfigurar red.

- **Una crónica para cada situación a identificar:** detección de pérdida, anormal retraso , etc.
- **Ejemplo de situación:** detectar pérdida de datos o notificación de congestión (ECN),
- **La crónica es:**

```
Message packet [ ?num_loss, ?seqn]{
}
Chronicle loss [?num_loss]
{
    occurs ((3,3), packet [ ?num_loss,?seqn], (t1,t2))
    ?seqn>?num_loss
    t1<t2
when recognized {
    emit event(packet_loss,t2);
}
}
```

Crónicas para auto-adaptar estrategias en Sistemas de Comunicación

- El mecanismo de auto-adaptación es guiado por el mecanismo de reconocimiento de situaciones basado en una maquina de estados
- **Mecanismo de auto-adaptación:**
 - **Maquina de estados de mecanismos de transporte** (TFRC o TCP Friendly Rate Control, TD – TFRC o Time-constrained and Differentiated TFRC, MP – TFRC, Multipath TFRC) y
 - **Transiciones según reconocimiento de crónicas,**

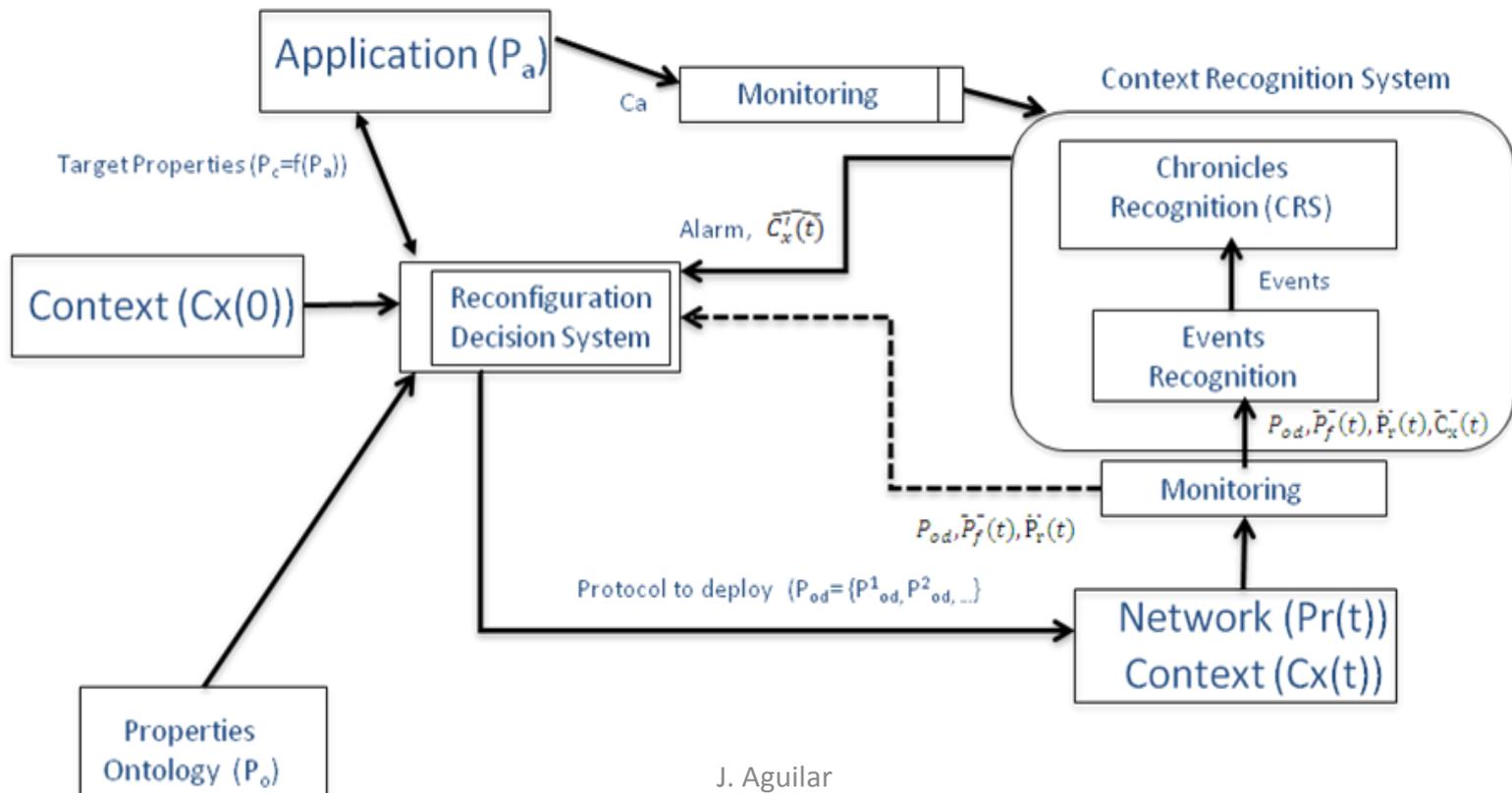


Diagnosis for Adaptive Strategies in collaborative Systems

- **Reconocer situaciones con crónicas**
 - En termino de diagnostico: síntoma-causa
 - Diagnostico « en línea »
- **Mecanismo de aprendizaje** para adaptar los protocolos a un contexto
 - Aprendizaje de la base de crónicas
 - Aprendizaje de los parámetros de las crónicas
- **Análisis de diagnosticabilidad** para garantizar la coherencia y completitud de la base de crónicas
 - Verificación de propiedades del tipo « crónicas exclusivas »,
 - Discriminar todas las situaciones de interés

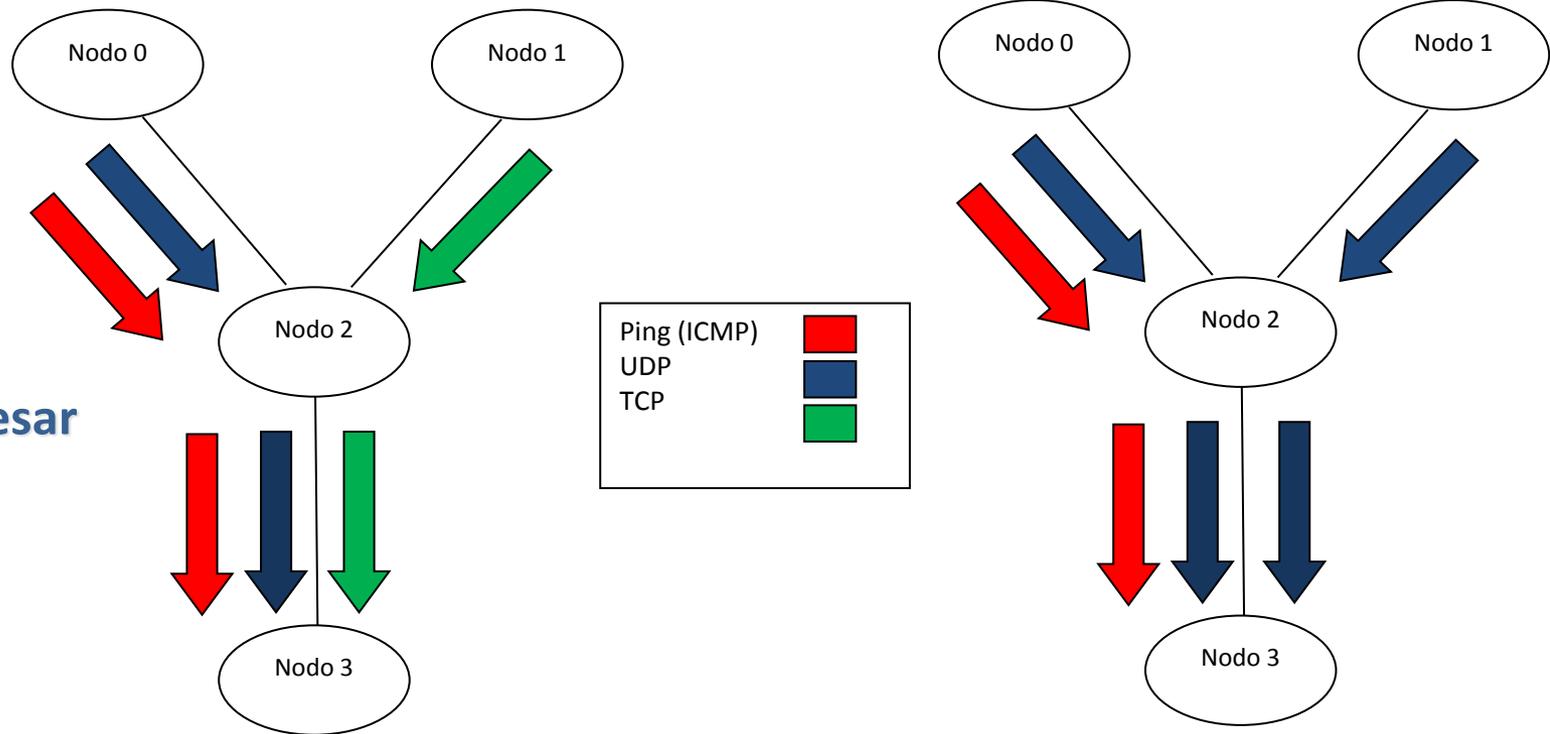
Diagnosis for Adaptive Strategies in collaborative Systems

Cx: Network context (network constraints + traffic constraints)
 Alarm: it is an indication to do a reconfiguration
 Ca: application + Collaboration context
 C'x: recognized context
 Pa, Pc, Po, Pod, Pf, Pr: properties



Diagnosis for Adaptive Strategies in collaborative Systems

1. Simular situaciones de ongestión a modelar



Preprocesar

Tres congestiones en cada simulación

Preprocesar

2. Extraer descriptores : *retardo* , *%Perdida*

✚ Script de calculo para ventanas de 10 paquetes transmitidos

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]
Action	Instant	Nsrc	Ndest	Type	Taille	Flag	IDflux	@src	@dest	N°Seq	IDpaquet	Timestamp

Archivo de trazas de NS-2

Diagnosis for Adaptive Strategies in collaborative Systems

2. Extraer descriptores : *retardo* , *%Perdida*

Script de calculo para ventanas de 10 paquetes transmitidos

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]
Action	Instant	Nsrc	Ndest	Type	Taille	Flag	IDflux	@src	@dest	N°Seq	IDpaquet	Timestamp

Archivo de trazas de NS-2



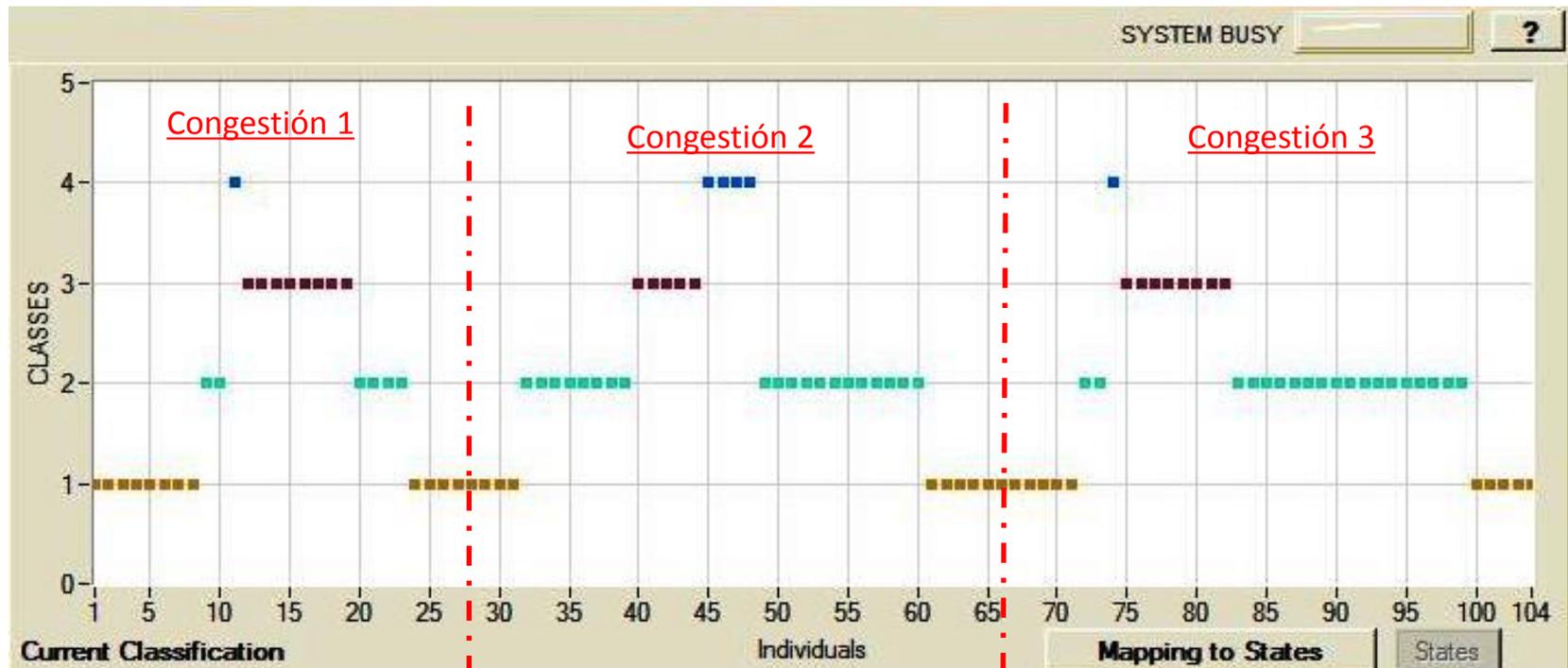
Resultado:

[0]	[1]
N° Individu	Instant

[0]	[1]	[2]	[3]
gigue	Délai	Débit	% perte

Archivo de resultados

Diagnosis for Adaptive Strategies in collaborative Systems



Secuencia N°1:

- E1 (C1,C2)
- E2 (C2,C3)
- E3 (C3,C2)
- E4 (C2,C1)

Secuencia N°2:

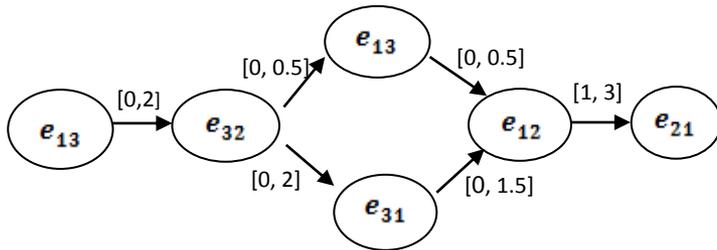
- E1 (C1,C2)
- E2 (C2,C3)
- E5 (C3,C4)
- E6 (C4,C2)
- E4 (C2,C1)

Secuencia N°3:

- E1 (C1,C2)
- E7 (C2,C4)
- E8 (C4,C3)
- E3 (C3,C2)
- E4 (C2,C1)

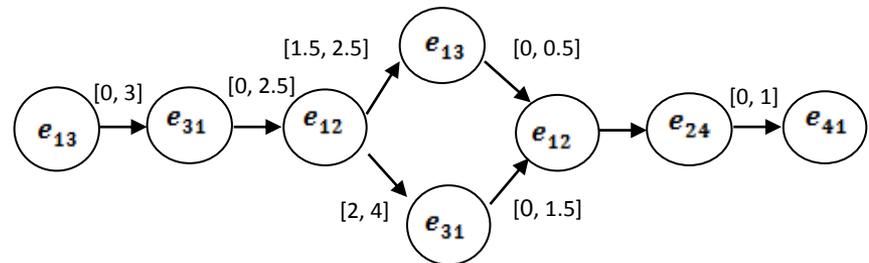
Especificar patrones temporales (crónicas)

Construir crónicas:



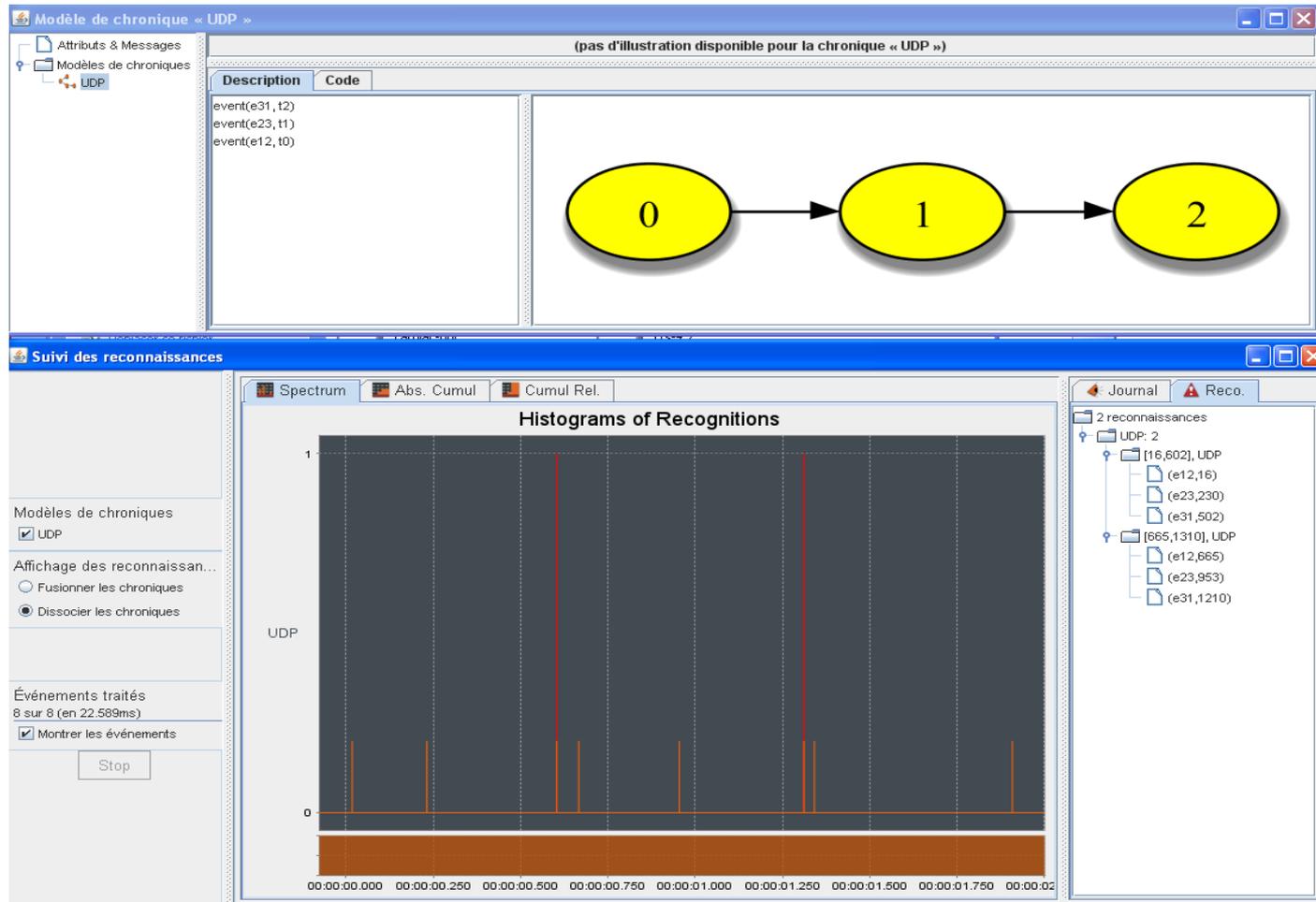
Congestión

Perdida de paquetes



Especificar patrones temporales (crónicas)

Resultado: *flujo1 : UDP/UDP*

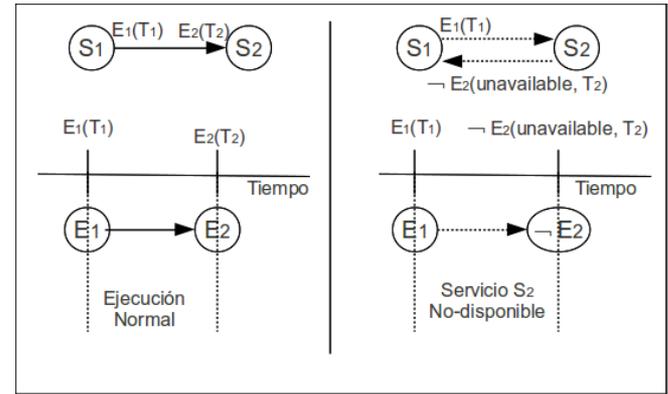


Aplicaciones SOA tolerantes a fallas

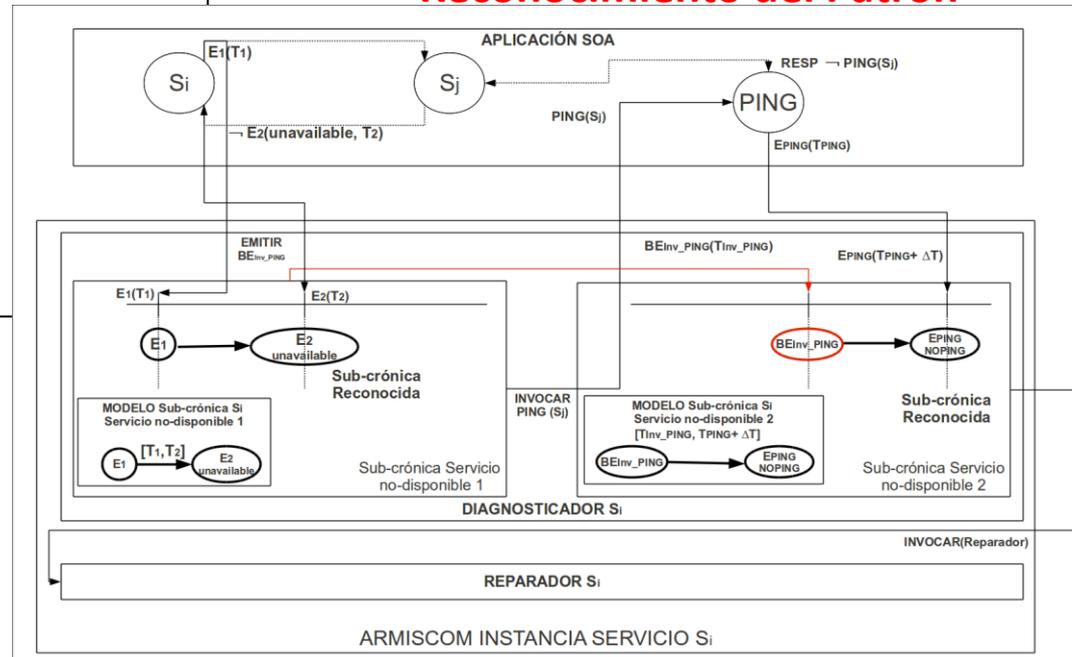
Diagnoser D_i	Diagnoser D_j
<p>Subchronicle S_i Servicio no-disponible 1 {</p> <p>Events{</p> <p>event(E_1, T_1)</p> <p>event($E_2, \text{Status:'unavailable' } T_2$)</p> <p>}</p> <p>Constrains{</p> <p>$T_2 > T_1$</p> <p>}</p> <p>When recognized{</p> <p>PING invoke($S_i, \text{hostname: } S_j$)</p> <p>emit event($BE_{invPING}, T_{invPING}$) to S_i diagnoser</p> <p>}</p> <p>}</p>	
<p>Subchronicle S_i Servicio no-disponible 2 {</p> <p>Events{</p> <p>event($BE_{invPING}, T_{invPING}$)</p> <p>event($E_{PING}, \text{status:'NOPING' } T_{PING}$)</p> <p>}</p> <p>Constrains{</p> <p>$T_{PING} > T_{invPING} + \Delta T$</p> <p>}</p> <p>When recognized{</p> <p>Repair invoke($S_i, \text{'Unavailable service } S_j$)</p> <p>}</p> <p>}</p>	

Patrón de Crónica Distribuida

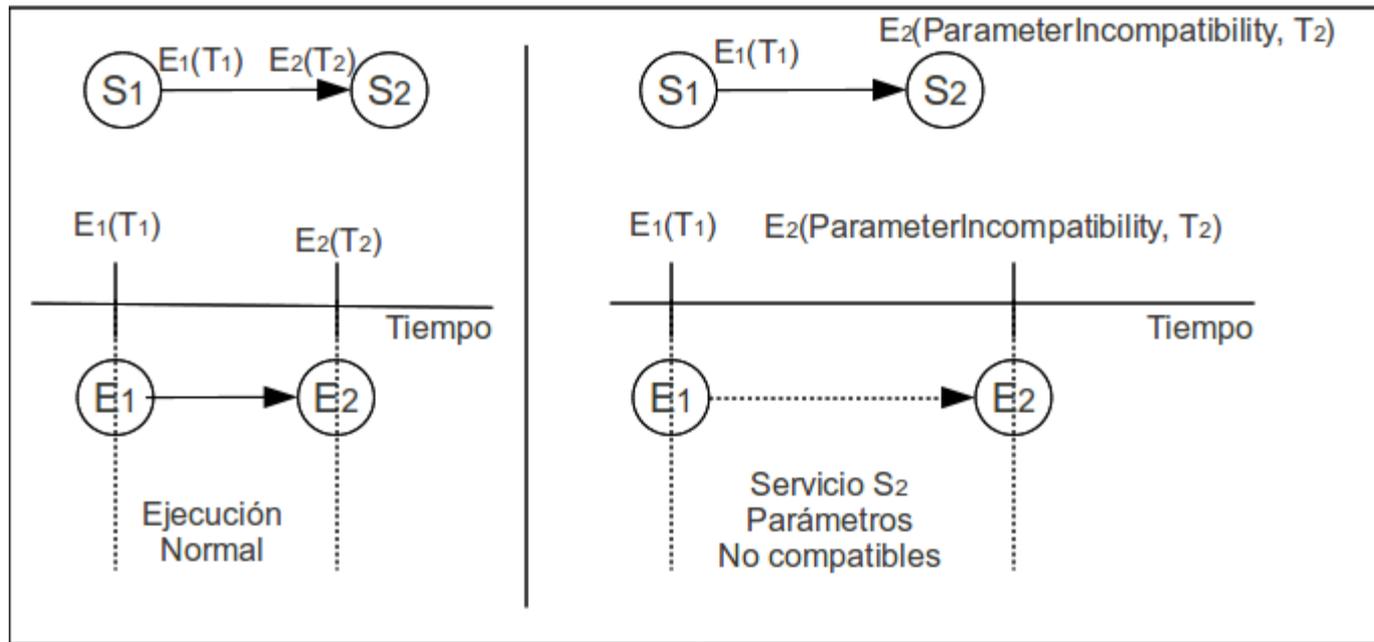
Falla de servicio no disponible



Reconocimiento del Patrón



Aplicaciones SOA tolerantes a fallas

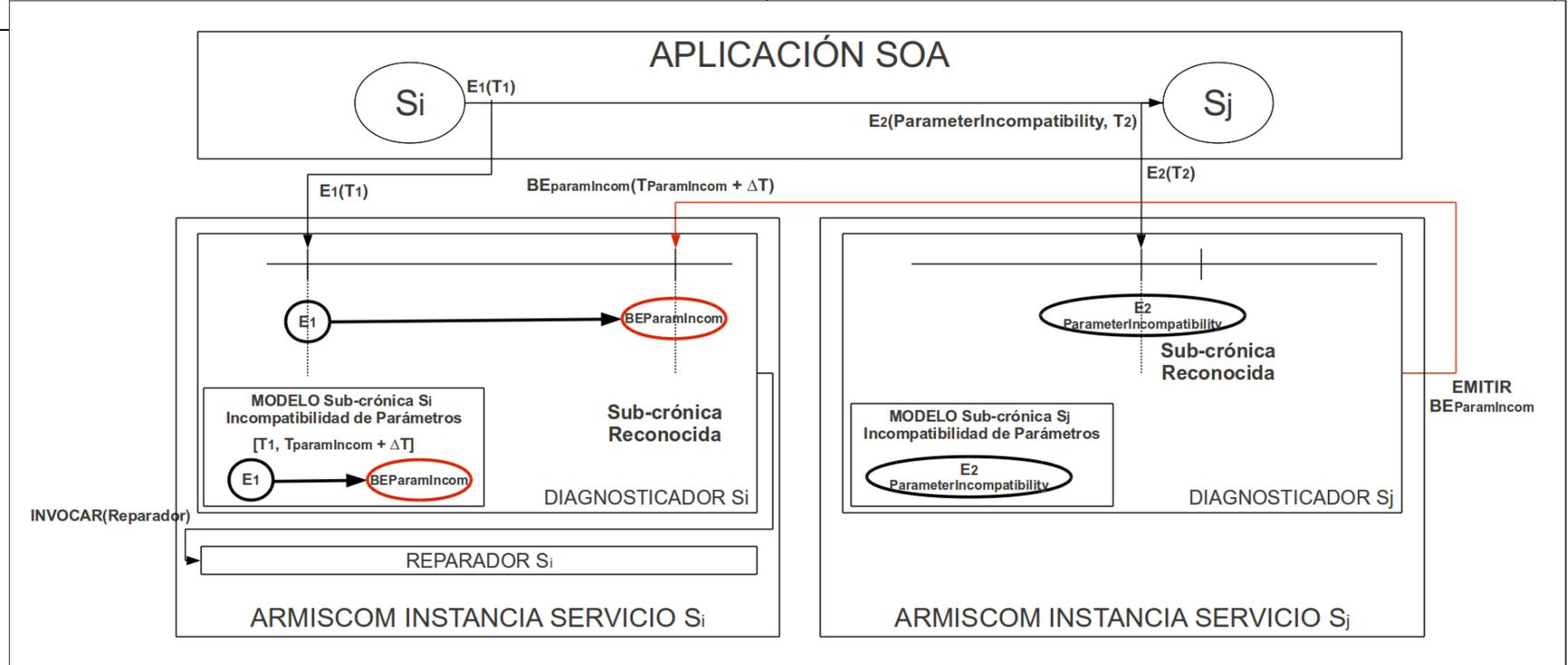


Falla de Incompatibilidad de Parámetros

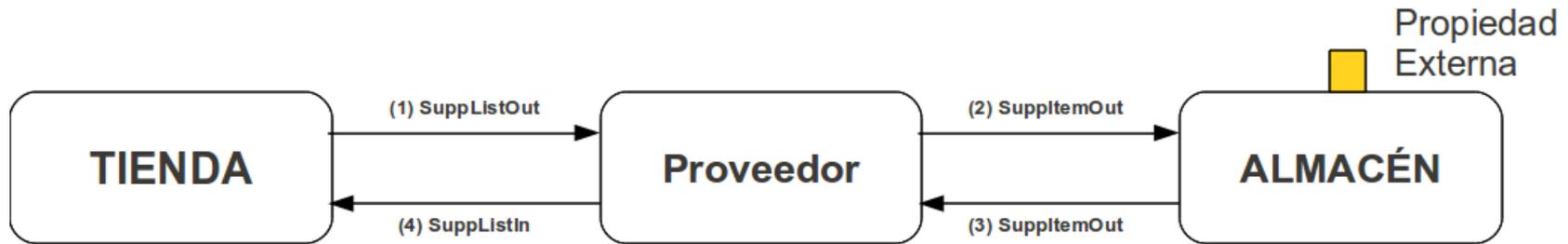
servicio se invoca con valores y/o tipos de datos de los argumentos incorrectos, con respecto a los tipos y a las restricciones definidas en el documento WSDL

Aplicaciones SOA tolerantes a fallas

<p>Diagnoser D_i</p> <p>Subchronicle S_i Incompatibilidad de Parámetros { Events{ event(E_1, T_1) event($BE_{ParamIncom}, T_{ParamIncom}$) } Constrains{ $T_{ParamIncom} > T_1 + \Delta T$ } When recognized{ repair invoke($S_i, 'Parameter Incompatibility'$) } }</p>	<p>Diagnoser D_j</p> <p>Subchronicle S_j Incompatibilidad de Parámetros { Events{ noevent($E_2, (0, T_2 - 1)$) event($E_2, Status:'ParameterIncompatibility', T_2$) } Constrains{ } When recognized{ emit event($BE_{ParamIncom}, T_{ParamIncom}$) to S_i diagnoser } }</p>
---	---



Aplicaciones SOA tolerantes a fallas



Tienda: es la tienda donde los usuarios compran los productos.

Proveedor: ofrece productos a la tienda; necesita comprobar en su almacén.

Almacén: es donde los productos se almacenan. Este proceso tiene un acuerdo de servicio (SLA) con el Proveedor, que consiste en que al menos un producto de la lista debe ser devuelto. Puede realizar búsquedas en sitios externos ExternalSearch (búsqueda Externa).

composición de servicios clásica de esta aplicación:

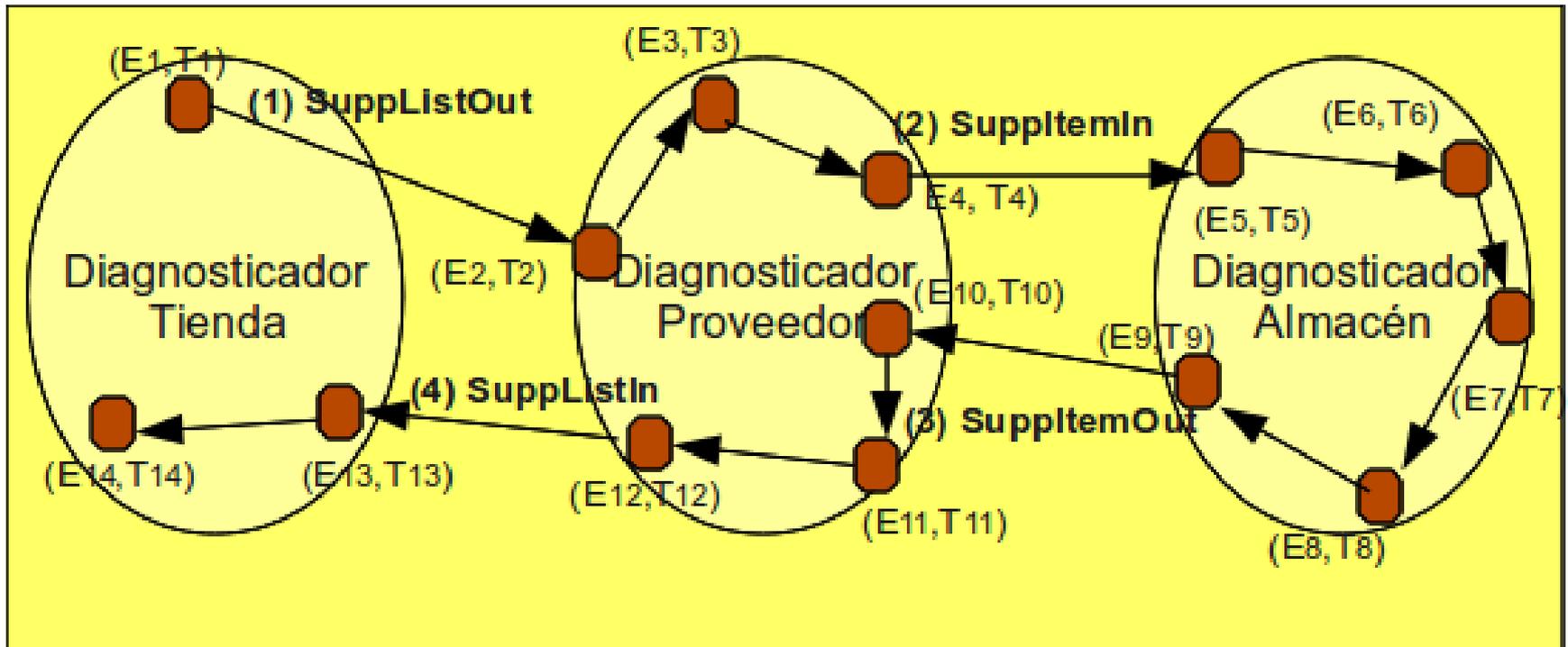
(1) SuppListOut: La tienda envía la lista de los productos que requiere al Proveedor.

(2) SuppltemIn: El Proveedor comprueba la disponibilidad de los productos en su depósito invocando al Almacén.

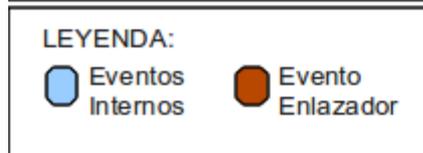
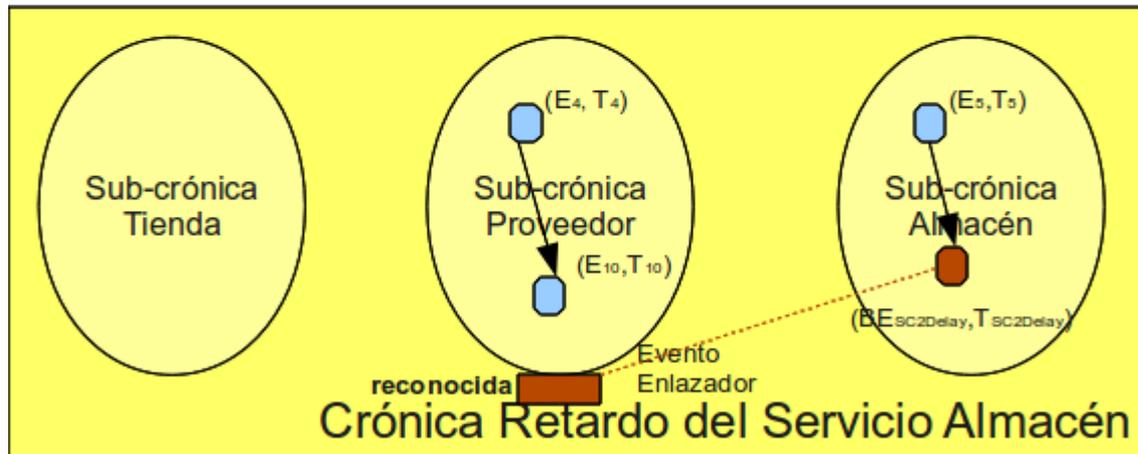
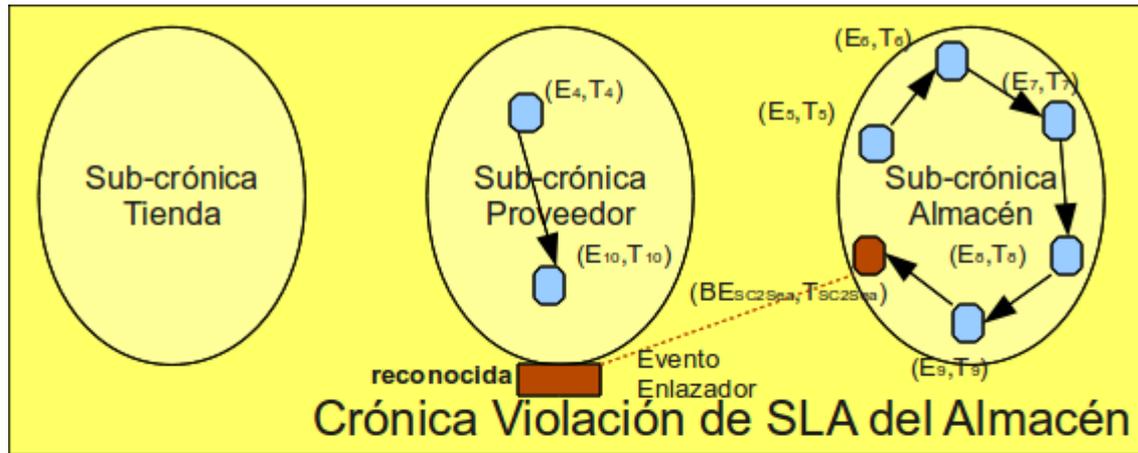
(3) SuppltemOut: El Almacén proporciona la respuesta sobre la lista de productos en el depósito al Proveedor, el cual debe contener al menos un producto.

(4) SuppListIn: El Proveedor notifica a la Tienda de los productos que puede ofrecer.

Aplicaciones SOA tolerantes a fallas



Aplicaciones SOA tolerantes a fallas



Crónica para violación de SLA por parte del servicio Almacén (caso de la operación Buscar Productos)

<pre> Sub-chronicle Tienda- BuscarProductos { Events{ } Constraints{ } When recognized{ } } </pre>	<pre> Sub-chronicle Proveedor- BuscarProductos { Events{ event(E₄: pl > 0, T₄), event(E₁₀: pl = 0, T₁₀) } Constraints{ T₁₀-T₄ ≤ 9 } When recognized{ Emit event(BE_{SC2Sea}, T_{SC2Sea}, Diagnoser 3) } } </pre>	<pre> Sub-chronicle Almacén- BuscarProductos { Events{ event(E₅; pl > 0, T₅), event(E₆: pl = 0, T₆), event(E₇: pl = 0, T₇), event(E₈: pl = 0, T₈), event(E₉: pl = 0, T₉), event(BE_{SC2Sea}, T_{SC2Sea}) } Constraints{ T₆-T₅ ≤ 1 T₇-T₆ ≤ 2 T₈-T₇ ≤ 1 T₉-T₈ ≤ 1 T_{SC2Sea}-T₉ ≤ 1 } When recognized{ repairer Invoke(Warehouse, 'SLA- Violation' } } </pre>
--	---	---

Agente Gladiador

Agente que juega COLISEUM de forma inteligente

- este juego consiste en un duelo entre dos personajes, el agente y un jugador, los cuales lucharán a muerte en la arena usando espadas.
- Durante el juego aparecerá una botiquín de forma aleatoria: conseguir la curación que él proporciona es un objetivo importante del juego

Crónicas

Relaciones

1-Short(d,e)

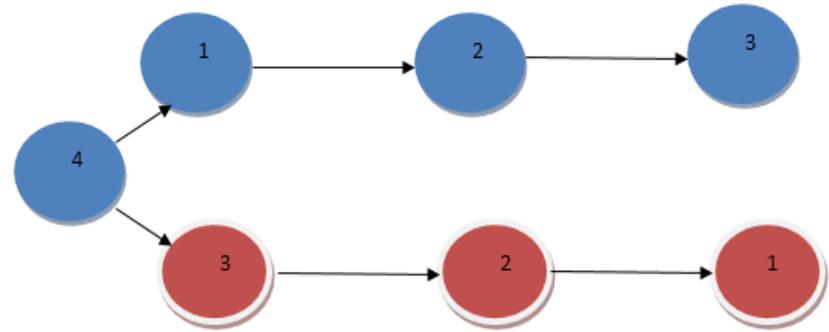
2-Mid(d,e)

3-Long(d,e)

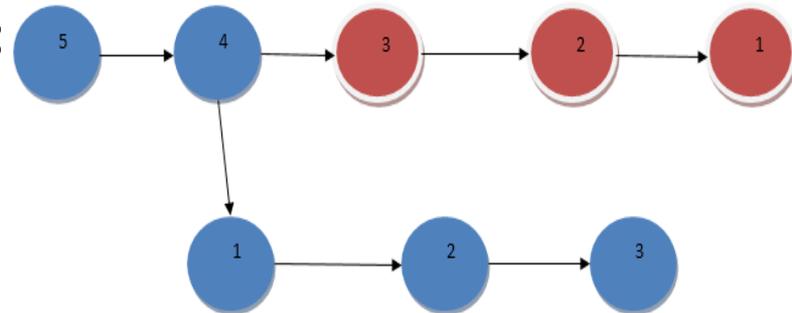
4-lowHealth(g)

5-avaLife(B)

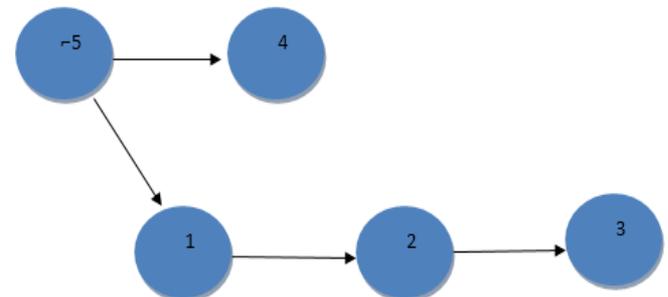
Atacando



Recuperando:



Debilitado



Sentencias en Prolog de las Crónicas

Lógica de Predicados de Primer Orden

- $\neg \text{lowHealth}(y,t_0) \wedge (\text{short}(y,z,t_1) \wedge \text{mid}(y,z,t_2) \wedge \text{long}(y,z,t_3)) \wedge (\text{long}(x,y,t_1) \wedge \text{mid}(x,y,t_2) \wedge \text{short}(x,y,t_3)) \wedge (t_0 < t_1 < t_2 < t_3) \Rightarrow \text{atacando}(y)$
- $\neg \text{avaLife}(B, t_0) \wedge \text{lowHealth}(y,t_1) \wedge (\text{short}(x,y,t_1) \wedge \text{mid}(x,y,t_2) \wedge \text{long}(x,y,t_3)) \Rightarrow \text{Debilitado}(y)$

Prolog

```
atacando(PosAgenteX, PosAgenteY, PosRivalX, PosRivalY, PosVidaX, PosVidaY, PosAgente2X,  
PosAgente2Y, PosRival2X, PosRival2Y, PosVida2X, PosVida2Y, PosAgente3X, PosAgente3Y,  
PosRival3X, PosRival3Y, PosVida3X, PosVida3Y, VidaRival) :- not(lowHealth(VidaRival)),  
long(PosAgenteX, PosAgenteY, PosRivalX, PosRivalY), mid(PosAgente2X, PosAgente2Y, PosRival2X, PosRival2Y),  
short(PosAgente3X, PosAgente3Y, PosRival3X, PosRival3Y), short(PosRivalX, PosRivalY, PosVidaX, PosVidaY),  
mid(PosRival2X, PosRival2Y, PosVida2X, PosVida2Y), long(PosRival3X, PosRival3Y, PosVida3X, PosVida3Y).
```



Modelos de Predicción: regresión, series temporales, etc.

Jose Aguilar

Pregunta

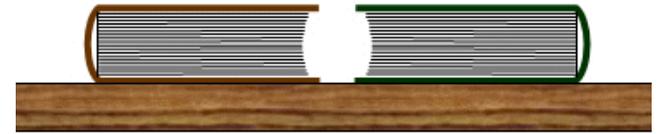


Se dispone de dos libros con muchas hojas, por ejemplo, dos guías de teléfono.

Se ponen frente a frente los dos libros y luego, con mucha paciencia, se intercalan las hojas de ellos.

Una vez que se intercalaron todas las hojas, se intenta separarlos tirándolos desde sus respectivos lomos.

¿Se pueden separar con facilidad?



plantee una predicción al respecto.

Aplicando fuerzas en los lomos respectivos de los libros,

¿será fácil o difícil separar los libros una vez que tienen mezcladas las hojas?

Predicción

- Predice un valor de una variable dada, sobre la base de los valores de otras variables, suponiendo un modelo lineal o no lineal de dependencia.
- **Ejemplos:**
 - Predecir las ventas de nuevos productos basados en gastos de publicidad.
 - Predecir la velocidad del viento como una función de la temperatura, humedad, presión de aire, etc.
 - Predecir comportamiento en el tiempo de los índices bursátiles (series de tiempo).

Modelos de Predicción

Las predicciones ayudan a la toma de decisiones en una gran variedad de areas.

- **Planificación y Control de Operaciones :** Las empresas usan predicciones para decidir que producir, cuando y donde.
- **Mercadeo:** Decisiones de precios, de gastos en publicidad, ...dependen fuertemente de las previsiones que se tengan sobre como van a responder las ventas a los diferentes esquemas de marketing.
- **Economía:** Predicciones de las variables macro-económicas claves como el PNB, Paro, Consumo, Inversión, Tipos de Interés, etc... son usadas por el gobierno para fijar su política monetaria y fiscal.
- **Financiera:** actores de los mercados financieros tienen un gran interes en la predicción de los rendimientos de activos financieros (acciones, tipos de interés, tipos de cambio, etc...).
- **Demografía:** La predicción de la población es crucial para planificar el gasto publico en sanidad, infraestructuras, educación, etc.

Modelos de Predicción

Piensa en una variable que quieras predecir. ***Que necesitas?***

- **Objeto a predecir:** Una serie temporal, un suceso, ...etc.
- **Formato de la Predicción:** Puntual, Intervalo, Densidad, ...etc.
- **Horizonte de la predicción:** Corto, Medio o Largo Plazo
- **Conjunto de Información:** Univariante o Multivariante
- **Metodos y Complejidad:** Modelos, ...etc.

Modelos de Predicción

Hay muchas formas de hacer predicciones; pero todas ellas tienen en común los siguientes ingredientes:

- 1. que hay ciertas regularidades que captar*
- 2. que tales regularidades son informativas sobre el futuro*
- 3. están encapsuladas en el método seleccionado para predecir*
- 4. se excluyen las no-regularidades*

Los principales métodos son:

- **Adivinación**
- **Extrapolación**
- **Encuestas**
- **Modelos de Series Temporales**

Evaluación de Predicciones

Al menos hay tres fuentes de error

- **Incertidumbre en la Especificación:** Todos los modelos están equivocados!!!!
(algunos mas que otros)
- **Incertidumbre en la Innovación:** Innovaciones futuras son desconocidas cuando se hace la predicción.
- **Incertidumbre en los Parámetros:** Los coeficientes que usamos para producir las predicciones son *estimaciones*, y por lo tanto están sujetas a variabilidad muestral.

Diferente medidas de errores de predicción

- **Error de Especificación**
- **Error de Aproximación**
- **Error de Estimación**

Evaluación de Predicciones

Las medidas mas comunes de la precisión de la predicción son:

Error cuadrático medio:
$$MSE = \frac{1}{T} \sum_{t=1}^T e_{t+h,t}^2$$

Raíz cuadrada del MSE
$$RMSE = \sqrt{\frac{1}{T} \sum_{t=1}^T e_{t+h,t}^2}$$

Error absoluto medio
$$MAE = \frac{1}{T} \sum_{t=1}^T |e_{t+h,t}|$$

donde $e_{t+h,t} = y_{t+h} - \hat{y}_{t+h,t}$ son los errores de predicción.

Modelos de Predicción

El modelo de regresión es un modelo explícitamente multi-variable, en donde la variable a explicar se explica y se predice en base a su propia historia pasada y la historia pasada de otras variables relacionadas.

$$y_t = \beta_0 + \beta_1 x_t + \varepsilon_t$$

$$\varepsilon_t \text{ es } WN(0, \sigma^2)$$

Predicción

Evaluación de la capacidad de predecir

- Dividir la muestra en dos partes; una para estimación del modelo y una para evaluar la capacidad de predecir.
- Estimar el modelo
- Calcular la predicción para los periodos no usadas.
- Comparar la predicción con valores reales (error del pronóstico)

Regresión

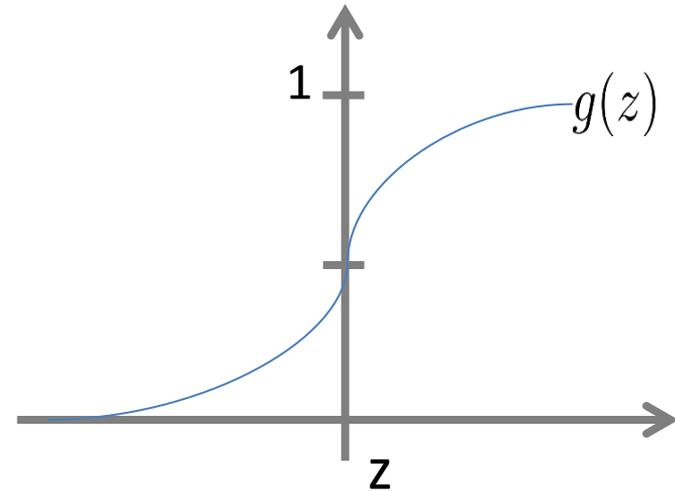
Regresión Lógica : $0 \leq h_{\theta}(x) \leq 1$

Cercano a clasificación

Regresión Lógica

$$h_{\theta}(x) = g(\theta^T x)$$

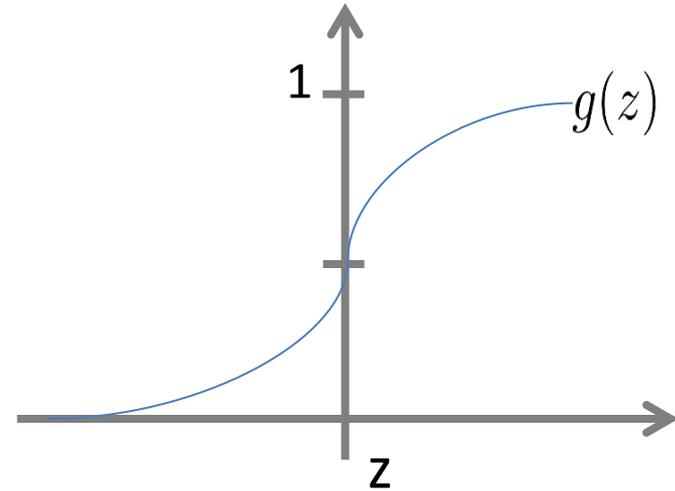
$$g(z) = \frac{1}{1+e^{-z}}$$



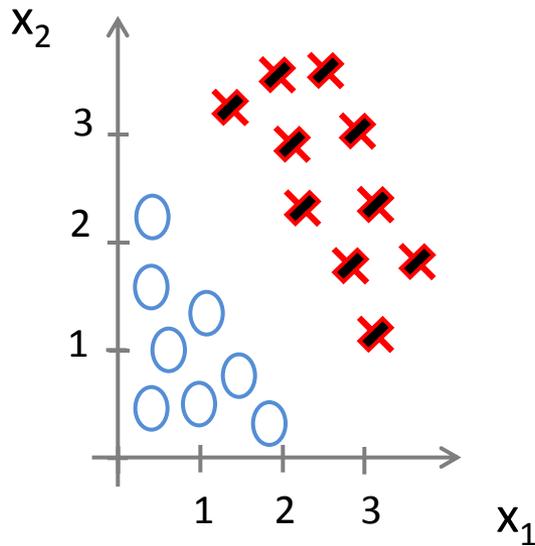
Regresión Lógica: Barrera de decisión

predice “y=1” si $h_{\theta}(x) \geq 0.5$

predice “y=0” si $h_{\theta}(x) < 0.5$

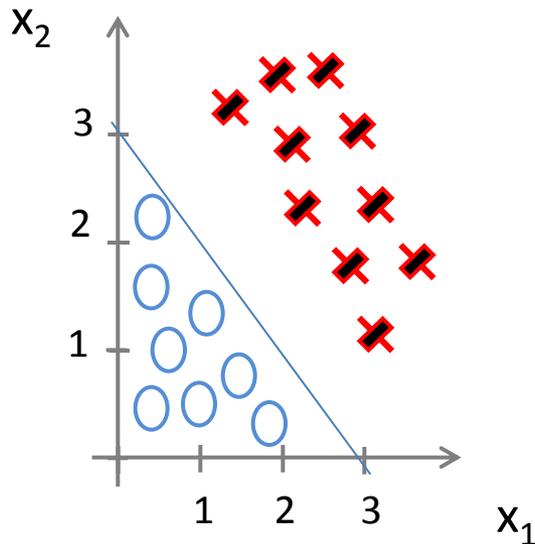


Regresión Lógica: Barrera de decisión



$$h_{\theta}(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

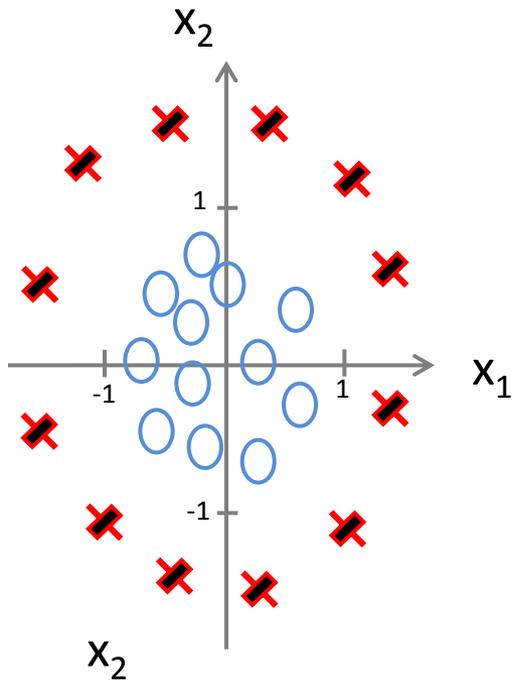
Regresión Lógica: Barrera de decisión



$$h_{\theta}(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

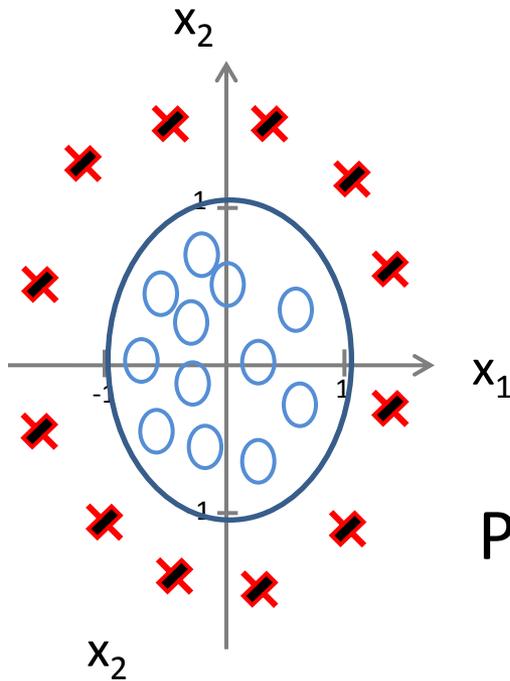
Predice "y=1" si $-3 + x_1 + x_2 \geq 0$

Regresión Lógica: Barrera de decisión



$$h_{\theta}(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2)$$

Regresión Lógica: Barrera de decisión



$$h_{\theta}(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2)$$

Predice "y=1" si $-1 + x_1^2 + x_2^2 \geq 0$

Regresión Lógica:

Entrenamiento y Función de costos

Conjunto de
entrenamiento

$$\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(m)}, y^{(m)})\}$$

:

n ejemplos

$$x \in \begin{bmatrix} x_0 \\ x_1 \\ \dots \\ x_n \end{bmatrix}$$

$$y \in \{0, 1\}$$

$$h_{\theta}(x) = \frac{1}{1 + e^{-\theta^T x}}$$

Como escoger el parámetro θ ?

Regresión Lógica: Función de costos

$$J(\theta) = \frac{1}{m} \sum_{i=1}^m \text{Cost}(h_{\theta}(x^{(i)}), y^{(i)})$$

$$\text{Cost}(h_{\theta}(x), y) = \begin{cases} -\log(h_{\theta}(x)) & \text{if } y = 1 \\ -\log(1 - h_{\theta}(x)) & \text{if } y = 0 \end{cases}$$

$$\min_{\theta} J(\theta)$$

Gradiente descendiente

$$J(\theta) = -\frac{1}{m} \left[\sum_{i=1}^m y^{(i)} \log h_{\theta}(x^{(i)}) + (1 - y^{(i)}) \log (1 - h_{\theta}(x^{(i)})) \right]$$

Queremos $\min_{\theta} J(\theta)$:

Repeat {

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta) \quad (\text{simultaneamente actualizar } \theta_j \text{)}$$

}

Función de costos

Regresión lineal: $J(\theta) = \frac{1}{m} \sum_{i=1}^m \frac{1}{2} (h_{\theta}(x^{(i)}) - y^{(i)})^2$

$$\text{Cost}(h_{\theta}(x^{(i)}), y^{(i)}) = \frac{1}{2} (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

Problema de Optimización!!!

Dado θ , queremos calcular

- $J(\theta)$

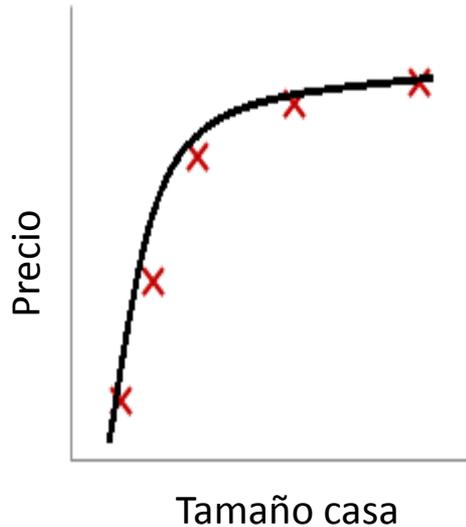
- $\frac{\partial}{\partial \theta_j} J(\theta)$

(for $j=0, \dots, n$) {

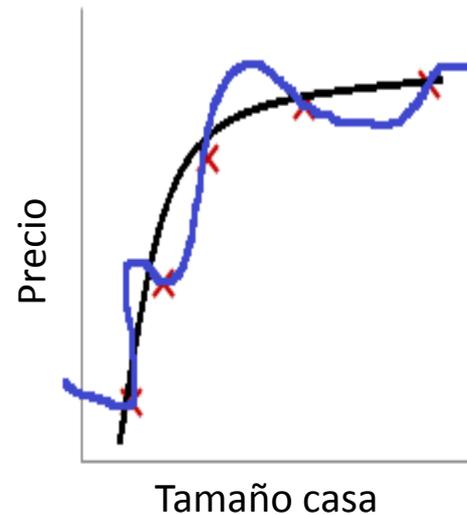
$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$

}

Función de costo



$$\theta_0 + \theta_1 x + \theta_2 x^2$$



$$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

Penalizar $\theta_3 \theta_4$

$$\min_{\theta} \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

Patrones Secuenciales

- Descubrir patrones en los cuales la presencia de un conjunto de ítems es seguido por otro ítem en orden temporal.
- Ejemplo: Encontrar y predecir el comportamiento de los visitantes de un sitio Web con respecto al tiempo.

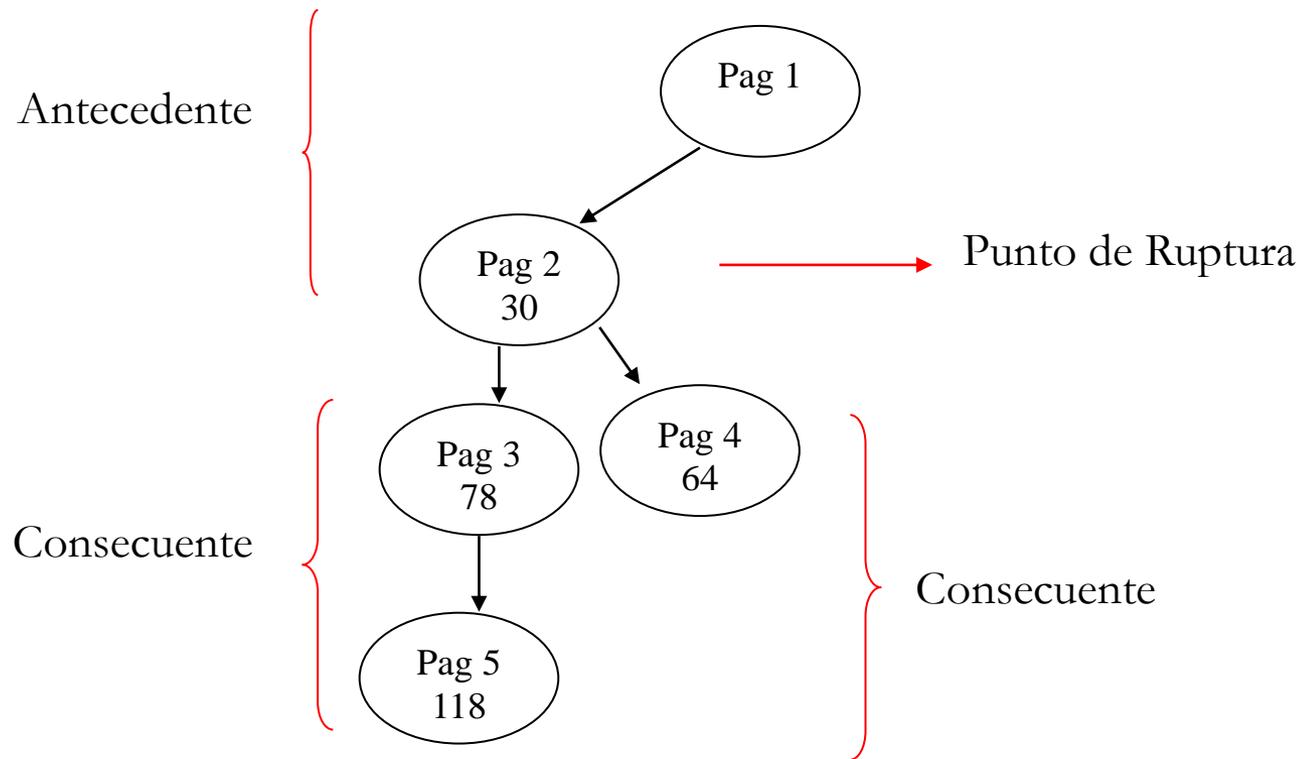
$[x1 \rightarrow x2 \rightarrow x3] \rightarrow [y1 \rightarrow y2]$ en t días

`[/public/team.jsp ->
/public/findUsers.jsp->
/private/mycourses/website/folders/assignment/assignment_view.jsp->
/public/portalDocument.js`

en 2 días

Patrones Secuenciales

Generación FBP-Árbol (Matriz FTM, Lista de Caminos)



Patrones Secuenciales

Algoritmo Patrones (FBP-Arbol, soporte, confianza)

La confianza de una *regla de comportamiento-frecuente* se representa como $conf(PIND \rightarrow PDEP)$ y define la probabilidad de recorrer el camino PDEP una vez se ha recorrido el camino PIND.

- Se recorre el árbol desde las hojas al nodo raíz.
- Teniendo en cuenta el soporte de cada camino las reglas son calculados como sigue.
- Buscar en hojas el punto de ruptura.
 - Si la hoja no es Punto ruptura, ir a hoja anterior.
 - Si la hoja es Punto Ruptura, calcular confianza.
 - Si $conf > confianza$, genera Patrón
 - Si $conf < confianza$, podar rama de árbol.

Resumen

- Para realizar una predicción es recomendable buscar y analizar la información disponible.
- Para realizar una predicción es necesario argumentarla, independientemente de que dichos argumentos sean correctos o erróneos.
- Es necesario poner a prueba la validez de la predicción.
- Hay que idear un procedimiento experimental, o teórico, para validar la predicción.
- Hay que estar dispuesto a modificar la predicción cuando la evidencia experimental arroja un resultado diferente al propuesto.